



UNIVERSIDAD POLITÉCNICA DE MADRID

FACULTAD INFORMÁTICA

**Arquitecturas de comunicaciones para la
computación algorítmica en poblaciones de
bacterias multi-cepa**

TESIS DOCTORAL

Ángel Goñi Moreno

Ingeniero en Informática

Año 2010

DEPARTAMENTO DE INTELIGENCIA ARTIFICIAL
FACULTAD INFORMÁTICA

**Arquitecturas de comunicaciones para la
computación algorítmica en poblaciones de
bacterias multi-cepa**

TESIS DOCTORAL

Autor: Ángel Goñi Moreno, Ingeniero en Informática.

Director: Juan Castellanos Peñuela, Doctor en Informática.

Año 2010



Tribunal nombrado por el Magfco. y Excmo. Sr. Rector de la Universidad Politécnica de Madrid, el día ____ de _____ de 2010.

Presidente: _____

Vocal: _____

Vocal: _____

Vocal: _____

Secretario: _____

Suplente: _____

Suplente: _____

Realizado el acto de defensa y lectura de la Tesis el día ____ de _____ de 2010 en la Facultad de Informática.

EL PRESIDENTE

LOS VOCALES

EL SECRETARIO

A M^a Teresa Moreno Grande, mi madre.

Una raya en el agua.

Agradecimientos

Mis sinceros agradecimientos:

Al Dr. Juan Castellanos Peñuela, mi tutor, por su dedicación, enseñanzas y amistad que espero dure mucho tiempo.

Al Dr. Martyn Amos por haber compartido conmigo sus conocimientos y por sus sugerencias y consejos.

A mis amigos por las risas y valores compartidos.

A mi guitarra por hacerme un sitio en la Música e influirme positivamente en todo lo que hago.

A Susana por alentarme con su magia y sus pinceladas de colores. Así puedo con todo.

Resumen

Esta Tesis establece su desarrollo en el área interdisciplinar de la biología sintética y, más concretamente, la computación con bacterias. La unión entre la biología y las ciencias de la computación tiene su raíz a mediados del siglo XX, siendo la biología una importante fuente de inspiración para desarrollar diferentes paradigmas de cómputo en analogía directa a los seres vivos. Sin embargo, no ha sido hasta finales del mismo siglo y principios del XXI cuando el flujo de inspiración cambió de rumbo y se empezaron a construir dispositivos moleculares que actuaran como rudimentarios computadores desempeñando tareas de cálculo lógico definidas. La suma de otra disciplina, la ingeniería, ayuda a abordar el diseño de estos biosistemas como una tarea formalizada de configuración de componentes. Al observar y entender una bacteria como un sistema cuya maquinaria está formada por un conjunto de piezas funcionales diferentes, surge el objetivo de alterar los mecanismos naturales de las bacterias con el fin de construir sistemas vivos con funcionalidades no naturales. Los algoritmos aquí especificados están diseñados para llevarse a cabo en comunidades de bacterias formadas por más de una cepa, para lo cual la presente Tesis propone arquitecturas de comunicaciones diversas que ayuden a la sincronización necesaria entre bacterias distintas, basándose para ello en las capacidades y mecanismos de comunicación que las bacterias muestran en estado natural, como son la conjugación bacteriana y el *quorum sensing*. Esta Tesis propone la modificación y manipulación de estos mecanismos para conseguir computaciones con rudimentarios sistemas de toma de decisiones y que, en un futuro, puedan servir al desarrollo de aplicaciones en campos científicos tan diversos como la medicina o la ecología. Entre los ejemplos de cómputo a los que se someten las nuevas arquitecturas diseñadas caben destacar problemas complejos (TSP, SAT) y un oscilador poblacional en el que una comunidad heterogénea de bacterias muestra oscilación única. Es importante enfatizar que uno de los principales objetivos de la Tesis es la validación, tanto biológica (conocimiento experto) como computacional (simulación) de los modelos diseñados. Ya que la Tesis tiene carácter eminentemente teórico, se lleva a cabo un fuerte proceso de validación que asegure en un porcentaje muy alto el éxito de la -futura- experimentación en laboratorio con estos diseños.

Abstract

The developments laid down in the present Thesis belong to an interdisciplinary research area called synthetic biology and, more specifically, bacterial computing. The link between biology and computer science have its roots in the middle of the 20th century when biology was an important source of inspiration in order to develop different computational paradigms in a straight analogy with living beings. However, it was at the end of the same century and the beginning of the 21st when the inspirational flows reversed and new molecular devices were design to behave like rudimentary computers able to perform basic logic calculations. The latest join of engineering sciences help researchers to deal with the design of biosystems from a formalised configuration of components point of view. Representing a bacterium in a schematic form like a microbial-scale machine whose system is composed of a group of different functional building blocks aims at modifying the natural mechanisms of a cell in order to achieve new living devices with non-natural functions. The algorithms specified here have been designed to be carried out using multi-strain bacterial communities where the coordination of the strain's actions conclude in the final behaviour. To achieve that goal, this Thesis proposes diverse communication architectures whose objective is to help the required synchronization between different functional bacteria based on the capabilities and natural mechanisms of communication that bacteria show in nature: bacterial conjugation and *quorum sensing*. The different proposals are based on the modification of those metabolic routes in order to obtain computations with elementary decision making processes which could be applied in the future in disparate disciplines like medicine or ecology. Among the computational problems tackled in the thesis and solved using the newly designed architectures there are hard combinatorial problems (TSP, SAT) and a population-based oscillator whose behaviour leads a multi-strain community to show a single oscillation signal. It is important to highlight the validation control of the models as one of the biggest objectives of the Thesis. Due to the essentially theoretical aspect of the Thesis, a strong biological validation (expert knowledge) as well as a strict computational validation (simulation) are carried out in order to assure a high probability of success in the -future- wet lab implementations of the proposed models.

Índice general

I	Introducción	VIII
1.	Document structure	1
2.	Visión general de la computación bacteriana	4
II	Computación molecular algorítmica	7
3.	Computación molecular <i>in-vitro</i>	8
3.1.	Estructura del ADN y manipulación	8
3.2.	Modelos de computación con ADN	13
3.2.1.	Adleman	13
3.2.2.	Head y Rozemberg	15
3.3.	Algorítmica y optimización	17
3.3.1.	Codificación y autoevaluación	18
4.	Fundamentos de la computación molecular <i>in-vivo</i>	22
5.	Salida analógica en la computación con bacterias	27
5.1.	Complejo Hin/ <i>hix</i> : antecedentes	29
5.1.1.	Resolución del BPP	30
5.1.2.	Resolución del HPP	32
5.2.	Definición del algoritmo celular propuesto	35

5.2.1. Diseño del biodispositivo	36
5.2.2. Simulación computacional	39
III Arquitecturas de comunicaciones	45
6. Preliminares	46
6.1. Procesamiento paralelo	48
6.1.1. Multiprocesamiento simétrico	49
6.1.2. Procesamiento masivamente paralelo	51
6.1.3. Procesamiento escalable	52
6.2. Paralelismo natural	53
7. Análisis biológico de la comunicación social bacteriana	57
7.1. Quorum sensing	59
7.2. Transferencia genética horizontal	64
7.3. Comunicaciones <i>Chip-to-Cell</i>	67
8. Arquitecturas de comunicación <i>in-vivo</i>	69
8.1. Arquitecturas basadas en señalización <i>QS</i>	70
8.1.1. Flujo de datos en sentido único	74
8.1.2. Autoseñalización	78
8.2. Arquitecturas basadas en la conjugación	80
9. Modelo de resolución del problema SAT	84
9.1. Células mínimas	86
9.2. Diseño del algoritmo	89
9.2.1. Formalización	93
9.3. Codificación genética	96
9.4. Computando una instancia del SAT	99
9.4.1. Discusión	103

10.Diseño de un biocircuito celular	106
10.1. Arquitectura del Modelo	106
10.2. Elementos estructurales: puertas lógicas celulares	108
10.3. Modelado del sistema	113
10.4. Simulación	119
10.4.1. Construcción del modelo	121
10.4.2. Resultados	127
10.4.3. Discusion	131
11.Diseño de un oscilador genético	133
11.1. Arquitectura del Modelo	133
11.2. Diseño de alto nivel	138
11.3. Diseño de bajo nivel	143
11.4. Simulación	148
11.4.1. Construcción del modelo	149
11.4.2. Resultados	156
11.4.2.1. Simulación de componentes	156
11.4.2.2. Simulación del sistema global	159
11.4.3. Discusion	174
IV Conclusiones y lineas de investigación futuras	177

Índice de figuras

3.1. Composición del nucleótido <i>Adenina</i>	9
3.2. Molécula de ADN	11
3.3. Computación de Adleman	15
4.1. Dogma biología molecular	23
4.2. Sectores de expresión	24
4.3. Inducción y represión	25
5.1. Solución <i>in-vivo</i> BPP. Codificación	31
5.2. Solución <i>in-vivo</i> HPP. Codificación	34
5.3. Salida analógica. Codificación	38
5.4. Salida analógica. Simulación	43
6.1. Concepto del procesamiento paralelo	49
6.2. Arquitectura de multiprocesamiento simétrico	50
6.3. Arquitectura del procesamiento masivamente paralelo	51
6.4. Arquitectura de procesamiento escalable	52
7.1. Quorum sensing. Arquitectura básica	60
7.2. Quorum sensing. Arquitecturas serie y paralela	62
7.3. Proceso de conjugación bacteriana	66
8.1. Arquitecturas básicas de comunicaciones <i>QS</i>	74
8.2. Diseño de la comunidad con función de dibujo de patrones	75

8.3. Formación de patrones en una comunidad bacteriana	77
8.4. Bandas de detección con expresiones lumínicas distintas	78
8.5. Consorcio celular con funcionalidad lógica AND	79
8.6. Esquema lógico de control de la conjugación	82
8.7. Inversor conjugativo	82
9.1. Manipulación automática con enzimas	92
9.2. Instrucción de evaluación	99
9.3. Algoritmo de resolución SAT	100
9.4. Codificación genética SAT	103
9.5. Flujo de datos en la comunidad SAT.	104
10.1. Puertas lógicas celulares	111
10.2. Modelo biocircuito XOR	116
10.3. Diseño de alto nivel del circuito simulado	121
10.4. Diseño interno de cada bacteria OR	123
10.5. Diseño interno de cada bacteria NAND	124
10.6. Diseño interno de cada bacteria AND	126
10.7. Salida biocircuito XOR final	129
10.8. Salida biocircuito XOR completa	130
11.1. Arquitectura con feedback	137
11.2. Secuencia de acciones de oscilación lumínica	143
11.3. Diseño interno de la bacteria central.	145
11.4. Diseño interno de las bacterias clientes	146
11.5. Modelado del umbral de sincronización	150
11.6. Modelado de la entrada manual	151
11.7. Diseño de alto nivel del oscilador	152
11.8. Diseño de la bacteria verde	153
11.9. Detalle de la expresión génica	154
11.10Diseño de la bacteria central	155
11.11Simulación de los operadores del promotor híbrido.	156

11.12Comportamiento de la bacteria servidora	157
11.13Salida total de la bacteria servidora	157
11.14Simulación de la bacteria servidora sin promotores híbridos	158
11.15Ruido producido por la bacteria servidora sin promotores híbridos . .	159
11.16Clients behaviour. a) Green client. b) Red client.	160
11.17Simulación estática sin promotores híbridos	162
11.18Simulación estática con promotores híbridos. Umbral alto.	163
11.19Simulación estática con promotores híbridos. Umbral bajo	164
11.20Simulación crecimiento común	169
11.21Simulación dinámica con promotores híbridos	170
11.22Simulación dinámica sin promotores híbridos	170
11.23Simulación con distinta degradación	171
11.24Función de crecimiento condicionada	173
11.25Simulación dinámica en situación de competición	174

Índice de cuadros

7.1. Bienes públicos	63
9.1. Detalle de codificación SAT: correspondencia genética	102
10.1. Reglas de inferencia de alto nivel	117
10.2. Reglas de inferencia de bajo nivel	119
10.3. Tabla de verdad del biocircuito XOR	128
11.1. Reglas de inferencia de alto nivel	141
11.2. Reglas de inferencia de bajo nivel.	147
11.3. Tabla de verdad de la bacteria servidora	156
11.4. Tiempos de generación	167

Parte I

Introducción

Capítulo 1

Document structure

First part: introduction

- Chapter 1: explicative index of the contents of the document.
- Capítulo 2: brief description of the relationship between biology and computer sciences. Union whose latest expression is synthetic biology and its subfield bacterial computing.

Second part: algorithmic molecular computing

- Chapter 3: detailed examination of some relevant *in-vitro* DNA computations and methods from its origins. It is emphasised in this chapter those aspects that represent a halfway point in the evolution from *in-vitro* to *in-vivo* computations. Moreover, an information encoding using DNA sequences suitable for algorithm optimization is proposed here.
- Chapter 4: analysis of the essential facts or principles which may be used as the starting point for bacterial computing modeling. The chapter pretends to offer a clear idea of the main tools available to alter or modify the natural metabolic routes *in-vivo*.
- Chapter 5: specific review of some recently build bacterial computers able to solve easy instances of hard combinatorial problems based on the Hin/*hix* protein complex found in at first in *Salmonella* bacteria. It is proposed in this

chapter a bacterial community designed to solve TSP using the same protein mechanism but displaying analogue reading in output responses instead of strict digital signals.

Third part: communication architectures

- Chapter 6: description of traditional parallel computer paradigms and comparison between conventional computers and bacteria. The results of that comparison and the analogies found will help to formalize the concepts used in next chapters.
- Chapter 7: analysis of bacterial social communication and its mechanisms. Both protocols, bacterial conjugation and quorum sensing, are detailed in this chapter in order to provide the reader with the necessary knowledge to tackle the following developments and designs.
- Chapter 8: brief summary of those recent works in bacterial computing aim at changing the communication mechanisms in order to suit specific objectives. This chapter describes the basics of their methodology and emphasises those parts susceptible of optimization.
- Chapter 9: proposition of a computational paradigm based on a multi-strain bacterial community able to solve an instance of the SAT problem autonomously. Bacterial conjugation is the point from which this design can achieve a solution of the problem by exchanging genetic information from one cell to another.
- Chapter 10: proposition of constructing a logic circuit with XOR functionality using as a foundation the quorum sensing mechanisms of prokaryotic cells. The *biocircuit* is a three-strain bacterial community where each strain shows a different logic behaviour: AND, OR and NAND. The emerging output achieved by engineering the communications pathways corresponds to the XOR behaviour expected.

- Chapter 11: proposal of a population-based oscillator based on an analogy with a server-client architecture using, as in the previous chapter, the quorum sensing mechanisms as the starting point for the design. The three strains that constitute this biodevice whose objective is to achieve a green-red oscillation are: 1) server bacteria, in charge of the synchronising the expression turns; 2) red client, which produces red light during its expression turn; 3) green client, which produces green light during its expression turn.

Forth part: conclusions

Conclusions extracted from the developments and proposals shown in previous chapters and analysis of future research directions obtained from this Thesis.

Capítulo 2

Visión general de la computación bacteriana

La computación con bacterias [Nor08] [Amo09] [Dan09] se considera una rama de la *biología sintética* la cual se define por medio de sus dos objetivos principales¹: 1) el diseño y construcción de nuevas piezas biológicas o biodispositivos y 2) la modificación de los sistemas existentes para desempeñar tareas definidas.

La relación entre la biología y las ciencias de la computación ha sido siempre muy estrecha desde el nacimiento de esta última. Aunque la computación molecular ha surgido como rama de investigación con entidad propia hace relativamente poco, la unión biología-computación no es para nada nueva.

Por un lado de la moneda, se puede señalar el papel fundamental que juega la informática en el avance de la biología gracias a la bioinformática. Durante años los biólogos han usado ordenadores para almacenar datos y analizar resultados experimentales, proceso que gracias a los avances de la informática cuenta con mayor precisión.

Así como lo anterior significa poner la informática al servicio de la biología ayudando a ésta a realizar su trabajo con mayor eficacia, el otro lado de la misma moneda trata de evolucionar los desarrollos, pensamientos y filosofías de las ciencias

¹<http://syntheticbiology.org>

de la computación utilizando los componentes biológicos como materia prima en lugar de electricidad, cobre y silicio.

La computación bacteriana o computación con bacterias se clasifica en esta segunda categoría y se fundamenta en la modificación de las rutas metabólicas naturales de las bacterias para recrear funcionalidades típicas de computadoras digitales con un dominio de aplicación radicalmente distinto. Así se aborda esta tarea gracias a la visión esquemática de una bacteria desde el punto de vista de la ingeniería, como un conjunto de piezas funcionales que forman un sistema. La variedad casi ilimitada de configuraciones de este sistema da lugar a esta línea de investigación.

La inspiración en la biología y los seres vivos por parte de los *padres* de la cibernética ha sido siempre una característica fundamental. Ya a mediados del siglo XX John von Neumann consiguió bosquejar una máquina de Turing autorreproductiva utilizando autómatas celulares [Neu66] basándose en una analogía con las células vivas. Von Neumann sostenía que el comportamiento de organismos naturales podía ser explicado por dispositivos similares, aunque aquellos fueran sistemas con mayor complejidad².

Siguiendo estos trabajos pioneros, se han realizado muchos esfuerzos por comprender los cálculos que la naturaleza realiza, y utilizar este conocimiento para conseguir mejores algoritmos e incluso nuevos tipos de computadores. Ejemplos bien conocidos de la aplicación del conocimiento obtenido de la biología son las redes de neuronas artificiales y los algoritmos genéticos o de forma más general la computación evolutiva. Todos estos modelos inspirados en la naturaleza se han agrupado bajo la denominación de *computación natural*.

El exponencial desarrollo de las técnicas de laboratorio y la optimización de la maquinaria usada han hecho posible que en los últimos años se aborde la computación natural no ya como una simple fuente de inspiración sino como una disciplina

²Se considera un sistema complejo cuando éste está compuesto por varias partes interconectadas o relacionadas de algún modo y cuyos vínculos contienen información adicional, esto es, como consecuencia directa de la relación entre las distintas partes de dicho sistema surgen nuevas propiedades denominadas emergentes.

de ingeniería práctica.

Mediante la lectura del resto de capítulos, el lector obtendrá una visión amplia del estado del arte actual en aquellos campos sobre los que tratan los desarrollos de esta Tesis Doctoral así como el conocimiento específico de los trabajos más relevantes en los inicios de esta disciplina.

Parte II

**Computación molecular
algorítmica**

Capítulo 3

Computación molecular *in-vitro*

Índice

3.1. Estructura del ADN y manipulación	8
3.2. Modelos de computación con ADN	13
3.3. Algorítmica y optimización	17

3.1. Estructura del ADN y manipulación

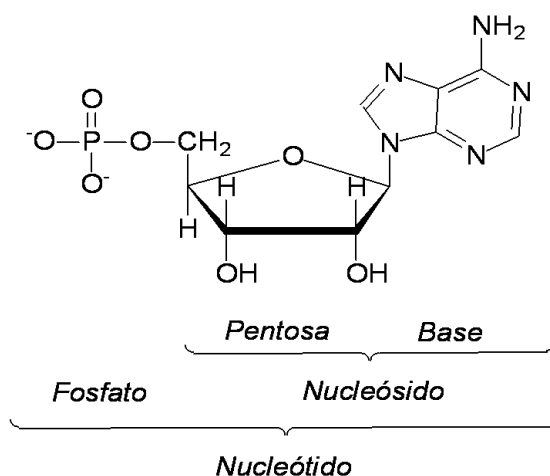
El nombre de ácido nucleico proviene de las sustancias ácidas que encontró Miescher [Mie75] en el núcleo de las células y que llamó inicialmente nucleína.

Los ácidos nucleicos son polímeros lineales en los que la unidad repetitiva es el nucleótido o monómero. Cada nucleótido está formado por: un azúcar (en concreto, una ribosa o desoxirribosa), una base nitrogenada y un ácido fosfórico. Se puede observar la formación de un nucleótido en la figura 3.1.

Los ácidos nucleicos pueden ser dos:

- *ADN o ácido desoxirribonucleico.*

Los monómeros que forman el ADN se llaman desoxirribonucleótidos. El nombre del azúcar en el caso del ADN es la desoxirribosa, al ser una pentosa tiene 5 átomos de carbono, el grupo de fosfato está unido al 5º carbono y la base al

Figura 3.1: Composición del nucleótido *Adenina*.

primero. Dentro de la estructura del azúcar hay un grupo hidróxido (OH) unido al tercer carbón. Las bases nitrogenadas que forman parte de los distintos nucleótidos del ADN difieren solo en la base, que pueden ser púricas (derivadas de la purina) como son la Adenina (figura 3.1 y la Guanina, o pirimidínicas (derivadas de la pirimidina) como son la Citosina y la Timina.

Ya que los nucleótidos difieren solo en las bases que portan, se les denota por la inicial de éstas, así hablamos de nucleótidos A, G, C y T.

- *ARN o ácido nucleico.*

Los monómeros que forman el ARN se llaman ribonucleótidos. Los ribonucleótidos difieren de los desoxirribonucleótidos principalmente en dos cosas. Primero, el azúcar de los ribonucleótidos es la D-ribosa, que tiene un grupo OH unido al segundo carbón en vez de un hidrógeno. Segundo, la base pirimidínica Timina es reemplazada por la base Uracilo, abreviada como U. En este caso las cuatro posibles bases son A, G, C y U.

Los ribonucleótidos con la base Adenina y con un triple grupo de fosfato se llama molécula de ATP, que es la principal fuente de energía de las células vivas.

Tanto las moléculas de ADN como las de ARN están formadas por largas se-

cuencias de nucleótidos consecutivos. Una diferencia fundamental es que la molécula de ADN es bicatenaria en su forma natural, es decir, está formada por dos cadenas de monómeros complementarias y unidas, mientras que la molécula de ARN es monocatenaria.

Es esta unión de cadenas complementarias la característica más relevante de las moléculas de ADN en referencia a su capacidad de cómputo. Tanto es así que los modelos de computación *in-vitro* que se verán más adelante basan su correcto funcionamiento en este proceso.

Existen dos tipos de enlaces entre nucleótidos. El primero es aquel que se origina entre dos nucleótidos con la finalidad de formar una misma hebra¹. En este caso podemos hablar de la dirección específica de la cadena dependiendo del extremo por el que se empieza a leer la hebra y el grupo carbono que tenga libre el azúcar de este extremo. De esta manera se identifica la dirección 5'-3' o la dirección 3'-5'.

En el segundo tipo de enlaces el nucleótido de una cadena se une al nucleótido de otra cadena con el fin de formar una molécula bicatenaria². Este enlace presenta la siguiente restricción en el emparejamiento de las bases: A siempre se une a T y C siempre se une a G, no siendo posible ninguna otra combinación. Cabe destacar como característica más relevante la mayor resistencia de la unión C-G frente a la unión A-T debido a que esta última presenta un puente de hidrógeno menos en el enlace.

Este principio de emparejamiento es llamado *complementariedad de Watson-Crick* [Wat53] y es la piedra angular para comprender la estructura y el funcionamiento del ADN. La unión de hebras complementarias da lugar a la forma de doble hélice característica de la molécula de ADN.

La figura 3.2 ilustra la composición de una molécula de ADN formada por

¹El grupo fosfato unido al quinto carbono de un nucleótido se une con el grupo hidróxido (OH) del tercer carbono del azúcar de otro nucleótido, formando un fuerte enlace covalente, llamado enlace fosfodiéster.

²La base de un nucleótido interactúa con la base de otro para formar un enlace de hidrógeno, que es un enlace débil.

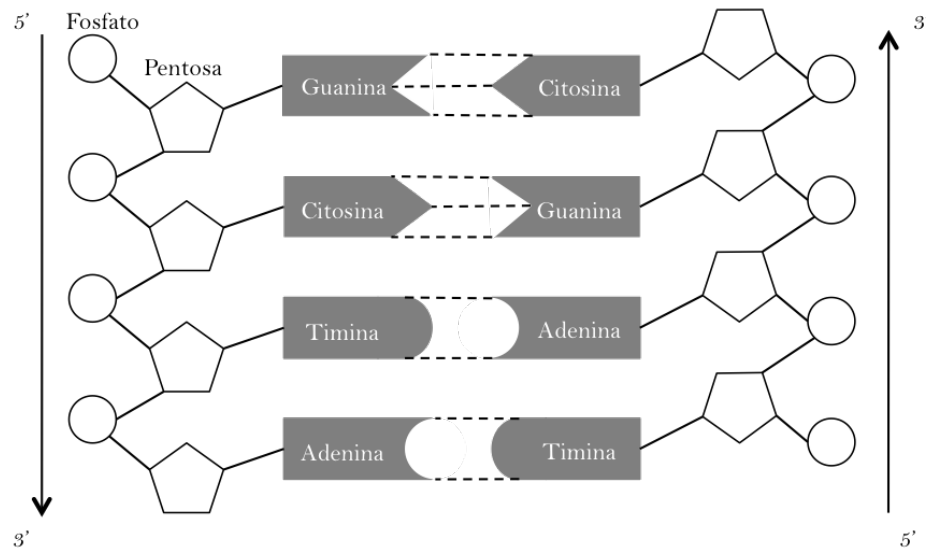


Figura 3.2: Molécula de ADN. Los pares G-C y A-T están formados por 3 y 2 puentes de hidrógeno respectivamente. En el extremo 5' de las hebras, el grupo fosfato está listo para unirse a un nuevo azúcar y en el 3' el azúcar listo para añadir otro grupo fosfato a la cadena.

las cadenas complementarias 5'-GCTA-3' y 3'-CGAT-5'. En ella se pueden observar los requisitos principales que caracterizan este tipo de moléculas: direccionalidad opuesta de las secuencias de nucleótidos, complementariedad en las uniones entre bases y número de puentes de hidrógeno que conforman estas uniones.

A cada *eslabón* de la cadena de ADN se le denomina *par de bases* (bp³). Así, la longitud de la cadena de la figura 3.2 es de 4bp.

Diversas técnicas permiten realizar las operaciones sobre moléculas de ADN en que se basan los modelos de computación descritos en la sección siguiente. De entre ellas, las técnicas consideradas como básicas se especifican a continuación:

Secuenciación de ADN. Este proceso consiste en la *lectura* de la información codificada en la molécula a estudio. Existen varios métodos para obtener la secuencia de nucleótidos del ADN como son el método de secuenciación automática y el método enzimático de terminación de cadena de Sanger [San77].

³Del inglés *base pair*.

Síntesis de nucleótidos. Proceso por el cual se producen muchas copias de la secuencia de nucleótidos específica que se desea obtener. La maquinaria necesaria para la síntesis de cadenas se perfecciona casi diariamente y las cantidades de bares de pares que se pueden sintetizar en una misma cadena aumentan año tras año. En un futuro cercano será posible incluso sintetizar, de una forma no muy costosa, el genoma entero de una célula primaria (formado por millones de bp).

Electroforesis. Es una técnica imprescindible en el manejo de ADN ya que permite determinar la longitud y tamaño de las cadenas. La técnica de la electroforesis en gel está basada en el hecho de que las moléculas de ADN están cargadas negativamente y al colocarlas en un campo eléctrico, se moverán hacia el electrodo positivo. Al ser la carga negativa de una molécula de ADN es proporcional a su longitud, la fuerza necesaria para mover la molécula también es proporcional a ésta y por tanto, las moléculas quedarán ordenadas según su dimensión.

Reacción en cadena de la polimerasa o PCR ⁴. Esta técnica permite producir millones de copias de una secuencia específica en un breve espacio de tiempo [Erl89], evitando así el uso bacterias como mecanismos naturales de replicación. Resulta una técnica muy útil en estrategias de clonación y detección cuando la cantidad de material es limitante así como en la genética de poblaciones.

Centrifugación. Se puede llevar a cabo un análisis de las características de las moléculas referidas no ya al peso o a la longitud sino a la secuencia específica de nucleótidos. Este análisis permitirá diferenciar codificaciones concretas en base a la densidad de la molécula. La técnica consiste en centrifugar la solución en la que se encuentren las moléculas a alta velocidad con el fin de crear un gradiente de densidad de la solución en el tubo de ensayo. Como consecuencia, las moléculas se distribuirán espacialmente diferenciándose por su propia densidad (ρ) la cual es directamente proporcional de la cantidad de C+G que contengan:

$$\rho = 1,66 + 0,098 \times \text{Frac}(G + C) \quad (3.1)$$

⁴Del inglés *Polymerase Chain Reaction*.

Desnaturalización. Mediante esta operación se consigue deshacer la estructura de doble hélice separando las dos hebras de la molécula de ADN original. El aumento de temperatura causa la destrucción de los puentes de hidrógeno que unen ambas cadenas.

Renaturalización o hibridación. Esta operación consiste en la restauración de la doble hélice a partir de moléculas monocatenarias. Al contrario que la operación anterior, hay que disminuir la temperatura con la finalidad de que se produzcan los procesos de hibridación.

Corte de moléculas. Es posible cortar las cadenas por sitios específicos dependiendo de las necesidades del algoritmo a implementar. Al ser la unión entre nucleótidos de la misma cadena mucho más fuerte que la formada por puentes de hidrógeno, no bastará con la aplicación de calor para deshacerla. Por ello, esta operación es llevada a cabo por unas proteínas denominadas enzimas de restricción las cuales se adhieren a las cadenas para cortarlas.

Unión de moléculas. Para que se produzca la unión entre dos moléculas bicatenarias es necesaria la actividad de la enzima llamada *ligasa*. Dependiendo del estado de los extremos (extremos cohesivos o romos) la unión será más o menos específica.

3.2. Modelos de computación con ADN

El objetivo de esta sección es describir dos modelos de computación *in-vitro* que por su importancia destacan en los inicios de la computación molecular.

3.2.1. Adleman

La construcción del primer computador de ADN en 1994 se debe a Leonard Adleman [Adl94] el cual fue capaz de resolver una instancia de un problema ma-

temático NP-Completo⁵ utilizando moléculas de ADN y valiéndose del conjunto de operaciones moleculares vistas previamente.

El problema que se resolvió era el HPP (*Hamiltonian Path Problem* o Problema del Camino Hamiltoniano) el cual pregunta, dado un grafo G dirigido con un vértice de entrada V_e y otro de salida V_s , si existe un camino hamiltoniano o no en el grafo. Un camino hamiltoniano es un recorrido del grafo que visita cada vértice exactamente una vez, comenzando en V_e y terminando en V_s .

La definición del grafo en que se basa el experimento, $G = \{V, A\}$, viene dada por el conjunto de vértices $V = \{V_0, V_1, V_2, V_3, V_4, V_5, V_6\}$ donde V_0 es el grafo de entrada y V_6 el de salida y el conjunto de aristas (de sentido único) $A = \{A_{0 \rightarrow 1}, A_{0 \rightarrow 3}, A_{0 \rightarrow 6}, A_{1 \rightarrow 2}, A_{1 \rightarrow 3}, A_{2 \rightarrow 1}, A_{2 \rightarrow 3}, A_{3 \rightarrow 2}, A_{3 \rightarrow 4}, A_{4 \rightarrow 1}, A_{4 \rightarrow 5}, A_{5 \rightarrow 1}, A_{5 \rightarrow 6}\}$.

El algoritmo exacto que usó Adleman se basa en crear, en una fase inicial, todas las soluciones posibles y posteriormente realizar procesos consecutivos de filtrado para determinar si existe o no una solución al problema. Es en ésta formación de todas las soluciones donde se aprovecha el paralelismo masivo moleculares que permite multitud de procesos simultáneos.

La definición de los datos de entrada en forma de secuencias de nucleótidos se realiza de la forma siguiente:

- Se asigna a cada vértice una cadena C de 20 nucleótidos. Por ejemplo, al vértice V_2 se le asigna la cadena $C(V_2) = TATCGGATCGGTATATCCGA$.
- Se crean las cadenas $C(V_2)_1 = TATCGGATCG$ y $C(V_2)_2 = GTATATCCGA$ correspondientes a la primera y segunda mitad del vértice V_2 respectivamente.
- Se denota por $\overline{C(V_i)_2 \cup C(V_j)_1}$ la arista $A_{i \rightarrow j}$.

Una vez sintetizadas todas las cadenas monocatenarias correspondientes a los vértices y las aristas, se vierten todas en un único tubo de ensayo. A continuación,

⁵Los problemas de esta clase de complejidad, sin entrar en detalles, necesitarán un tiempo exponencial para su resolución por un algoritmo exacto (no heurístico) lo cual los hace atractivos para su estudio en paradigmas de cómputo masivamente paralelos como la computación con ADN.

y como resultado de los procesos paralelos de hibridación entre secuencias, se obtendrá una amplia variedad de moléculas bicatenarias como la ilustrada en la figura 3.3.

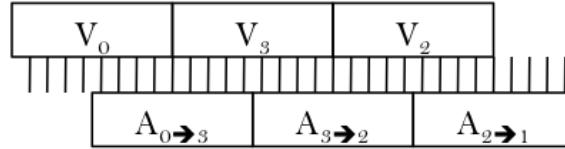


Figura 3.3: Molécula de ADN correspondiente al camino $V_0 \rightarrow V_3 \rightarrow V_2$

Después de la generación de todas las moléculas posibles hay que filtrar éstas con los criterios requeridos por el enunciado del problema HPP. Se realizaron los siguientes procesos de filtrado para la obtención de la solución:

1. Rechazar todos los caminos que no empiezan en V_0 y terminan en V_6 .
2. Rechazar todos los caminos que no contienen exactamente N vértices. Siendo $N = 6$.
3. Para cada vértice V_i , rechazar los caminos que no lo contengan (6 procesos de filtrado secuenciales).

La respuesta al problema es afirmativa si queda algún camino en el tubo después de los procesos de filtrado y negativa en caso contrario.

3.2.2. Head y Rozemberg

Se detalla en esta sección el experimento de Head y Rozemberg [Hea00] el cual representa un peldaño intermedio en el proceso de traducción de la computación *in-vitro* a *in-vivo* por la naturaleza del material empleado para llevar a cabo el algoritmo.

Este experimento resuelve el problema del Conjunto Independiente Máximo, también NP-Completo, utilizando estructuras más complejas de ADN como son los

plásmidos o vectores plasmídicos⁶. Dado un grafo $G = \{V, A\}$ siendo V el conjunto de vértices y A el conjunto de aristas definiéndose cada una por un par de vértices de la forma $A_i = \{V_j, V_k\}$, un subconjunto S de V se dice que es independiente si para cada arista A_i de G el subconjunto S no contiene sus dos vértices simultáneamente.

El grafo del problema a resolver se define por $V = \{V_0, V_1, V_2, V_3, V_4, V_5\}$ y $A = \{A_1, A_2, A_3, A_4\}$ con $A_1 = \{V_0, V_1\}$, $A_2 = \{V_1, V_2\}$, $A_3 = \{V_2, V_3\}$ y $A_4 = \{V_3, V_4\}$.

La codificación de los datos del problema en secuencias de ADN se realiza en relación a las características de las estructuras utilizadas: los plásmidos. Se diseña entonces un único plásmido que contendrá 6 sectores diferenciados (longitudes distintas) correspondientes a los seis vértices del grafo. Cada uno de estos sectores se sitúa entre dos enzimas específicas y particulares de cada sector que permiten la rápida eliminación (por corte de cadenas) de éstos en el plásmido que representa el conjunto total inicial.

Al inicio del algoritmo [Hea00] se tiene un tubo de ensayo T_0 con multitud de copias del plásmido inicial. A continuación se realiza un proceso de *borrado* por cada arista $A_i = \{V_j, V_k\}$ del grafo consistente en copiar el contenido de T_0 en dos nuevos tubos T_1 y T_2 y eliminar de los plásmidos de T_1 el sector que representa al vértice V_j y de los plásmidos del tubo T_2 el sector V_k . Al final de cada iteración se mezcla el contenido de los dos tubos T_1 y T_2 en un nuevo tubo inicial y se repite el proceso para la siguiente arista.

Como resultado y al final de todas las iteraciones se tendrá en un solo tubo todos los conjuntos mínimos del problema planteado. Por último, una sencilla operación de electroforesis puede determinar que plásmido representa el máximo de todos ellos.

Uno método más complejo pero más efectivo de lectura de los plásmidos finales [Hen05] consiste en la traslación proteica de los datos codificados. Esto es, convertir⁷

⁶Los plásmidos son moléculas de ADN circular normalmente presentes en las bacterias. Muchos de los desarrollos *in-vivo* actuales se fundamentan en el uso plásmidos como base de la codificación de información.

⁷Mediante la utilización de una enzima específica y la colocación de un *promotor* al comienzo

cada plásmido en una proteína y analizar ésta obteniendo resultados más exactos sobre la secuencia de nucleótidos que codificaba el plásmido original.

Son varios los ejemplos que, desde el experimento de Adleman, implementan computaciones con ADN *in-vitro* [Ben01] [Gua96] [Wan01].

3.3. Algorítmica y optimización

Con el objetivo de optimizar los modelos de computación molecular fundamentados en algoritmos exactos han sido desarrolladas varias alternativas tales como algoritmos heurísticos⁸ fundamentados en la computación evolutiva [Bac99] [Woo01] [Waka05]⁹ [Saka05].

Se hace énfasis en esta sección en la codificación característica de los datos del problema adecuada para la implementación de algoritmos genéticos (AG) los cuales son métodos de búsqueda adaptativa inspirados en la teoría de la evolución de las especies. Como resultado de su aplicación cada generación de resultados mejorará su adaptabilidad al medio (faceta dependiente de los requisitos del problema) con respecto a las siguientes generaciones.

Este tipo de algoritmos consta de cinco etapas elementales:

- Codificación del dominio. Los datos o individuos iniciales del problema, que conformarán la población inicial, se deben codificar utilizando el alfabeto compuesto por los símbolos A, C, G y T.
- Evaluación de la población. Los individuos de una población deben ser expuestos a un proceso de evaluación que determine cuál de ellos está mejor adaptado para cumplir los objetivos del problema en cuestión. Se llamará *fitness* al grado de adaptación de un individuo.

de la secuencia codificada. En el capítulo siguiente se detallará este proceso.

⁸Se denomina heurística a la capacidad de un sistema para realizar de forma inmediata innovaciones positivas para sus fines.

⁹Modelo de computación *in-vivo*.

- Selección. En esta etapa se seleccionan aquellos individuos mejor evaluados.
- Cruzamiento. El algoritmo de cruzamiento (aspecto fundamental en un AG) debe ser capaz de realizar el apareamiento entre dos individuos con el fin de crear uno nuevo de manera que sea aprovechada la información de ambos progenitores.
- Mutación. Como última etapa, y de manera aleatoria, pueden suceder eventos sobre los nuevos individuos formados en el paso anterior que modifiquen su codificación. La finalidad de este proceso es posibilitar la exploración acelerada del espacio de búsqueda del problema.

La aplicación iterativa de estas cinco etapas proporcionará soluciones al problema más exactas en cada generación. De esta manera se aborda la complejidad de los problemas aumentando la complejidad del mecanismo de procesamiento.

3.3.1. Codificación y autoevaluación

Uno de los mecanismos que otorga gran autonomía a los computadores moleculares es la autoevaluación de las soluciones. Se trata entonces de encontrar una codificación de datos adecuada para encriptar en cadenas de ADN información del fitness de los individuos.

Para trabajar con material genético como materia prima es importante conocer la inestabilidad de las moléculas de ADN y las reacciones impredecibles que pueden ocurrir. Por ello la primera tarea a llevar a cabo al comienzo del desarrollo de un computador molecular es la correcta codificación de la información para reducir al máximo las reacciones no deseadas.

Así, son muchas las condiciones fisico-químicas que pueden provocar la pérdida de nucleótidos o la rotura de una cadena y debido a la complementariedad se pueden producir uniones no deseadas entre los datos codificados (en moléculas monocatenarias) del problema a resolver.

También es importante notar que varias operaciones moleculares tales como la electroforesis en gel o la centrifugación isopícnica, las cuales son herramientas esenciales, están basadas en características concretas de la codificación de la cadena. Estas últimas características deben ser tenidas en cuenta cuando se afronta el problema de implementar algoritmos que optimicen los métodos exactos. Por ejemplo, para los algoritmos genéticos la electroforesis puede ayudar a determinar los individuos mejor adaptados en un problema en el que la longitud de la cadena sea la condición fundamental del fitness.

Se deben tener en cuenta todas las características anteriores para poder asegurar la estabilidad de los datos del problema así como su diferenciación individual. Se afrontan así dos problemas distintos aunque estrechamente relacionados: la codificación de lenguajes estables y la codificación específica de cada dato de entrada.

Codificación de lenguajes estables. Con la finalidad de codificar palabras estables sobre el alfabeto compuesto por los cuatro nucleótidos, se crean lenguajes formales [Kar05] [Cui07] denominados lenguajes libres de uniones¹⁰ que aseguran esta característica para cualquier palabra w del lenguaje.

Las principales variables a tener en cuenta en la definición de un lenguaje libre de uniones y que determinarán el grado de estabilidad son la longitud (d) de las palabras (w) y el número de estas ($|w|$) que contendrá el lenguaje. Si se quiere crear un conjunto de 20 palabras ($w = 20$) con 5 nucleótidos cada una ($d = 5$) la estabilidad referida a la disminución de uniones indeseadas será mayor que al crear un lenguaje con $w = 40$ y $d = 5$. Esto es debido a la existencia de sólo 4 símbolos en el alfabeto ($\{a, c, g, t\}$) de manera que a menor número de palabras, mayor será la distancia de Hamming¹¹ entre ellas.

El objetivo de un lenguaje libre de uniones es, aparte de asegurar el mismo grado de estabilidad a condiciones fisico-químicas de las palabras, alcanzar la máxima distancia de Hamming posible entre cualquier par de palabras del lenguaje.

¹⁰Del inglés *bond-free language*.

¹¹Se entiende aquí por distancia de Hamming entre dos palabras como el grado de complementariedad entre ambas.

Codificación específica de los datos. A diferencia de los modelos de computación molecular de filtrado como el experimento de Adleman visto anteriormente en los que los datos de entrada al problema (ciudades, caminos, ...) se codificaban en cadenas aleatorias, se propone aquí un modelo de codificación de datos basado en la caracterización de éstos.

Como caracterización se escoge la cantidad de bases G (guanina) y C (citosa) que contenga la cadena sirviéndose así de las funcionalidades de la operación de centrifugación. De esta manera cada dato contendrá la información de adaptación al entorno desde su codificación inicial, es decir, en lugar de codificar datos de manera aleatoria (partiendo de un lenguaje dado) se podrán codificar datos con cierta información particular de cada uno.

Se consigue así que la evaluación de los datos sea automática y simultánea al proceso de codificación de éstos. Mediante la centrifugación se realiza entonces el proceso de selección para elegir los mejor evaluados de una determinada población. Es en este punto donde la ecuación 3.1 diferencia la codificación de cada cadena para formar el gradiente de *fitness* en la población molecular para identificar de manera rápida y fácil los individuos seleccionados.

Para asegurar el correcto funcionamiento del paradigma de computación propuesto no basta con la codificación del *fitness* en los datos sino que se debe encontrar el equilibrio entre dicha caracterización y el uso de un lenguaje estable que asegure la reducción de la inestabilidad intrínseca de las moléculas de ADN.

Sin embargo, los lenguajes libres de uniones no permiten la caracterización de los datos del problema. Así, no se podría codificar un dato con una palabra de un lenguaje estable sin quitarle la capacidad de autoevaluación ya que sería imposible distinguir a ese dato de otro formado por el mismo lenguaje. Si esto fuera posible se estaría violando una de las principales restricciones de estos lenguajes: la formación de moléculas con el mismo grado de estabilidad independientemente de las condiciones ambientales o las operaciones moleculares aplicadas sobre ellas.

Por tanto los datos deben tener un formato tal que permitan la codificación

estable y la caracterización simultáneamente. Este formato consiste en la utilización de un lenguaje estable para definir la mayor parte de la secuencia dejando un porcentaje, a establecer dependiendo del tamaño del conjunto de datos, para codificar la caracterización o *fitness* de cada dato concreto. El formato es:

$$\text{Lenguaje estable } (w_1) \cup \text{Fitness } (w_2) \cup \text{Lenguaje estable } (w_3)$$

donde la unión de las tres secuencias w_1 , w_2 , y w_3 forma el dato final.

Un experimento como el de Adleman en el que se utilizara este modelo de codificación podría incorporar pesos en los caminos en forma de concentración de G+C de manera que al quedarse únicamente con los caminos hamiltonianos se pudiera identificar con un proceso de centrifugado cual de ellos representa el camino más corto. Además la tasa de errores debidos a la inestabilidad molecular decrecería considerablemente.

La codificación específica ayuda también a definir algoritmos de cruzamiento dirigidos, distintos del *splicing* natural o cruzamiento de un punto, de forma que la molécula resultante presente una evolución del grado de adaptación al medio que sea consecuencia de aprovechar la información codificada en las moléculas progenitoras.

Esta codificación de los datos en forma de estructuras caracterizadas como piezas individuales constituye un paso previo a la codificación de los datos del problema en genes para realizar computaciones *in-vivo*. En desarrollos propuestos en capítulos posteriores se modela la resolución de problemas combinatorios *in-vivo* utilizando bacterias y haciéndose corresponder a cada dato del problema con un gen específico.

Capítulo 4

Fundamentos de la computación molecular *in-vivo*

Aunque la idea de realizar computaciones a nivel molecular fue propuesta hace décadas [Fey61] [Hea87], su implementación real ha tenido que esperar a la optimización y desarrollo de las técnicas de laboratorio así como la maquinaria disponible. El previo estudio teórico desembocó en la computación molecular *in-vitro* al igual que ésta ha llevado a experimentar con formas de vida básicas como las bacterias, organismo que sirve como materia prima en los desarrollos de esta Tesis Doctoral.

Este capítulo describe los mecanismos básicos en que se fundamentan las operaciones realizadas *in-vivo* con el fin de proporcionar procesos mínimos cuya combinación de lugar a operaciones más complejas.

El dogma central de la biología molecular (figura 4.1) es el proceso por el cual a partir de ADN se genera ARN (ARN mensajero o mARN) y, a partir de éste, proteínas mediante procesos de transcripción y translación respectivamente. Se puede incluir en este dogma la operación de la replicación de ADN por la cual dos moléculas de ADN resultan de la duplicación de la información de otra molécula inicial.

Este enunciado es generalmente cierto, aunque algunos organismos presentan variantes particulares. Por ejemplo los retrovirus, entre los que se incluye el VIH o

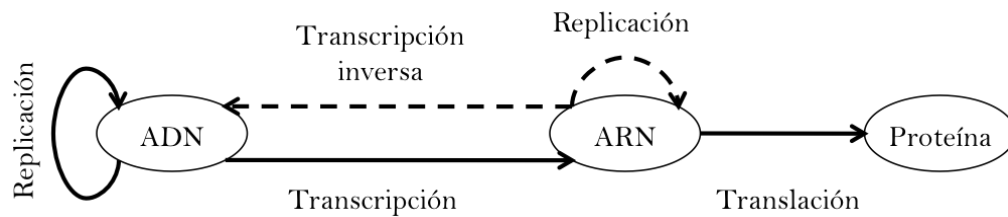


Figura 4.1: Dogma central de la biología molecular.

virus del SIDA, tienen la capacidad de invertir el proceso de transcripción y producir copias de ADN a partir de mARN. También hay otros virus que presentan la habilidad de replicar el ARN de manera directa sin necesidad de utilización de ADN.

Los bloques moleculares principales donde se representa la información genética se denominan genes estructurales, los cuales son secuencias de ADN que codifican proteínas específicas. Mediante los procesos de transcripción y traducción anteriormente vistos se extrae de ellos dicha información para producir proteínas que son los elementos moleculares activos.

El proceso de transcripción lo comienza una enzima particular llamada ARN polimerasa la cual se une a una secuencia particular llamada *promotor* que se sitúa al comienzo del gen indicando el lugar de comienzo de la transcripción (ver figura 4.2a). Esta enzima transcribe el segmento de ADN que se encuentre desde la secuencia promotora hasta una secuencia específica de terminación (*terminador*) dando lugar a la molécula monocatenaria de mARN.

De forma similar, la traducción se lleva a cabo gracias a complejos moleculares llamados ribosomas que se unen al mARN por un sitio específico denominado *RBS*¹ y procesan el mensaje, desde una secuencia de comienzo (tripleta AUG, ver figura 4.2b) hasta una de finalización, produciendo una secuencia de aminoácidos mediante el tARN o ARN de transferencia. Este proceso termina la expresión genética del gen estructural inicial.

El proceso de transcripción puede ser regulado por unas moléculas denomina-

¹Del inglés *Ribosome Binding Site*.

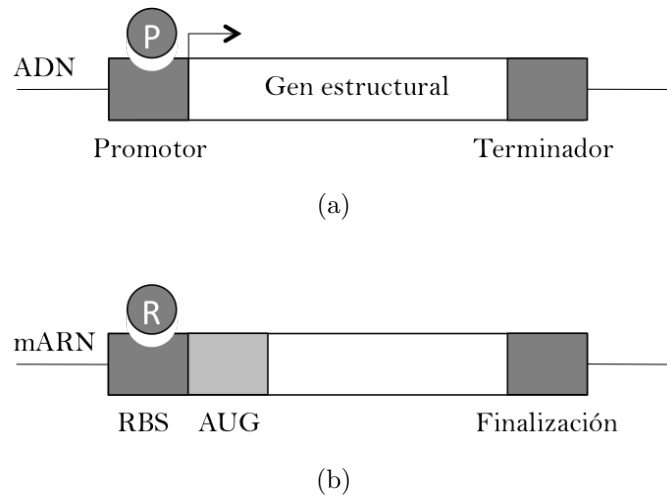


Figura 4.2: Principales segmentos involucrados en la expresión genética. a) Segmento de ADN; P: enzima polimerasa. b) Segmento de mARN; R: ribosoma.

das *factores de transcripción* con características de activación o represión dependiendo del efecto que produzcan: incremento o decremento, respectivamente, del nivel de transcripción. De esta manera es posible realizar un control sobre la expresión genética y proceder al *encendido* o *apagado* de genes específicos con el fin de recrear un circuito genético.

Se muestran así en la figura 4.3 diferentes maneras de utilización de esta especie de *circuitería* genética. En la figura 4.3a se muestra una situación en la que un promotor específico transcribe de manera normal el gen objetivo a no ser que el factor de represión se encuentre presente, momento en el cual la transcripción se bloquea. Se puede comparar la actividad de esta molécula represora al funcionamiento de un interruptor que deja o no pasar corriente eléctrica a un dispositivo.

Para recrear circuitos que cuenten con mayor número de variables de entrada se puede hacer uso de moléculas inductoras (figura 4.3c) que bloqueen la actividad del represor (de-represión) volviendo a dejar libre el promotor para que continúe la expresión del gen objetivo². En este caso, el valor de salida (la proteína resultante

²Este es el caso del operón *lacZYA* en la bacteria *E.Coli* controlado por el gen *lacI* cuya expresión genera una molécula que bloquea la transcripción de dicho operón a no ser que la presencia de lactosa sea suficiente para desactivar la funcionalidad de la molécula represora.

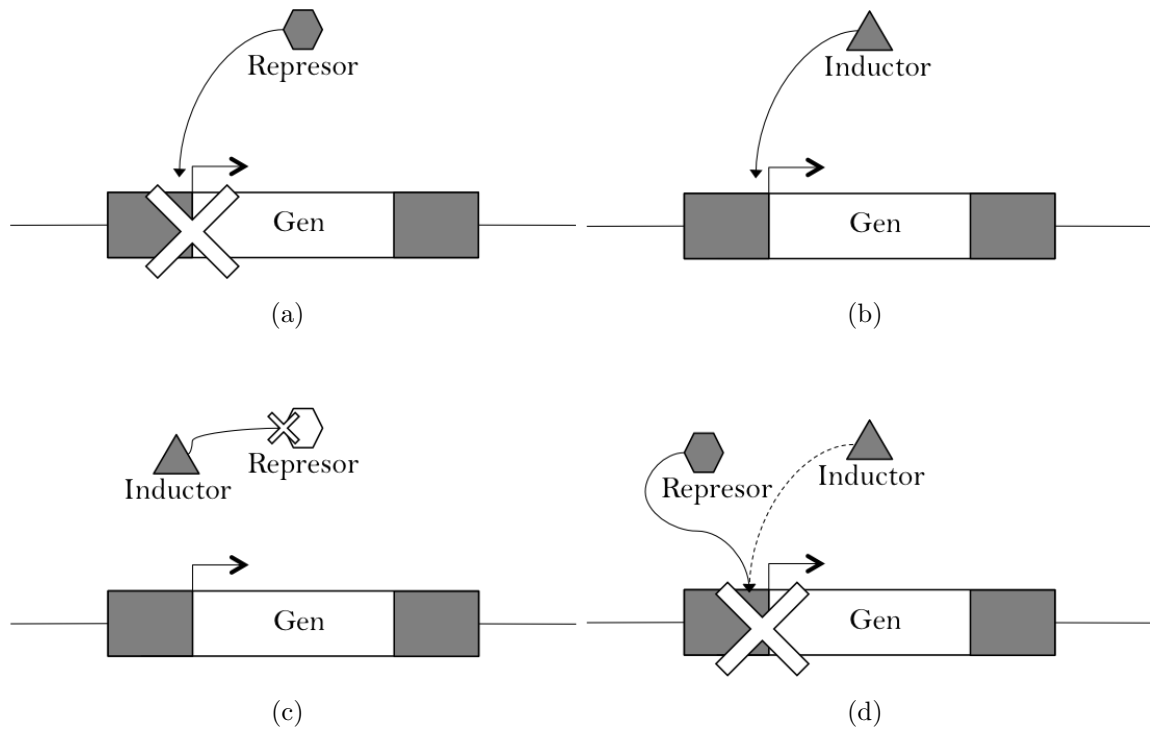


Figura 4.3: Formas de controlar la expresión genética mediante moléculas efectoras en distintas situaciones: a) Represión de la transcripción. b) Activación de la transcripción. c) De-represión. d) Bloqueo de la inducción.

de la expresión) dependerá del estado de dos variables.

La figura 4.3b representa una situación en la que el promotor por sí sólo no es suficiente para realizar la expresión y necesita la presencia de una molécula inductora que, en este caso, actúa directamente con él induciendo la transcripción. De forma análoga al caso anterior, se puede contar con la actividad de una molécula receptora que bloquee la actividad de la inductora (figura 4.3d).

La primera propuesta de crear circuitos lógicos en forma de redes genéticas [Wei99] utilizando las operaciones detalladas anteriormente y modificando las rutas metabólicas naturales de las bacterias, significó un importante impulso para los desarrollos dirigidos hacia computación *in-vivo*. Un año después, dos trabajos de vital importancia para la computación molecular reforzaron ese impulso.

El primero de esos trabajos [Elo00] describe la construcción de una red genéti-

ca que resultaba en un comportamiento oscilatorio (señal de reloj). Siguiendo la *programación* de esta red, cada bacteria producía una expresión lumínica intermitente³. La idea de la oscilación genética como medio de controlar la sincronización de procesos moleculares sirve de fuerte inspiración para uno de los desarrollos de esta Tesis Doctoral: el oscilador poblacional.

El segundo trabajo [Gar00], de publicación simultánea al anterior, describe la implementación de un *switch* en la bacteria *E.Coli* cuyo comportamiento se controla por estímulos externos a la bacteria. La implementación de operaciones lógicas utilizando material genético y modificando las rutas metabólicas naturales de las bacterias mostraba así sus comienzos.

³Cabe destacar un trabajo de reciente publicación [Dan10] en el que partiendo de estas bacterias con oscilación independiente se modela su colaboración para que funcionen todas al mismo tiempo.

Capítulo 5

Salida analógica en la computación con bacterias

Índice

5.1. Complejo <i>Hin/hix</i> : antecedentes	29
5.2. Definición del algoritmo celular propuesto	35

Este capítulo propone un modelo de computación celular cuya importancia radica no tanto en el proceso de la computación como en la forma de expresar las salidas de éste.

En lugar de expresar el éxito o fracaso de un algoritmo molecular mediante salidas que sean estrictamente digitales, como presencia/ausencia de expresión lumínica, se presenta aquí el modelado de una salida analógica. Esta permite notificar una cantidad de información superior utilizando el mismo mecanismo de señalización.

Si se utilizan señales fluorescentes, como es el caso que atañe al desarrollo de este capítulo, se puede hablar de salidas difusas en las que se definen distintos niveles de luminosidad. De este modo la salida no tiene por qué ser un simple *on* u *off* sino que la conclusión del algoritmo se expresaría de forma gradual.

El comportamiento analógico en las Ciencias de la Computación es un objetivo muy deseado debido a la mayor capacidad de expresión que proporciona respecto

a la computación digital estricta. Sin embargo, solo se consigue emular el comportamiento analógico mediante aproximaciones probabilistas ya que la materia prima es, al fin y al cabo, digital. Relativo a este campo de investigación hay que destacar el esfuerzo empleado por desarrollar puertas lógicas analógicas [Vig03] y circuitos digitales con comportamiento analógico [Loe09].

Las ayudas a la decisión en los dispositivos digitales que ayudan a la modelización de un comportamiento analógico simulado básico pueden observarse fácilmente en el caso de un inversor lógico. La decisión basada en la probabilidad guía el comportamiento de la puerta de la siguiente manera: en lugar de tener como entradas los valores digitales 0 ó 1, se tiene una probabilidad p de tener un 0 y una probabilidad q de tener un 1. Se cumple entonces que $p(0) + q(1) = 100\%$ estando ambas probabilidades sujetas a cambios por la acción de las salidas de otras puertas. Así, el circuito puede simular un comportamiento analógico que, sin embargo, no será real.

En los procesos metabólicos que tienen lugar en las bacterias, así como en el comportamiento de los diferentes agentes moleculares se pueden identificar fácilmente ambos paradigmas de cómputo: digital y analógico. En concreto, la expresión genética puede ser controlada de manera analógica para mostrar un amplio dominio de valores de expresión.

Es este último proceso de control el punto de partida para diseñar el desarrollo propuesto en este capítulo. Se da importancia aquí al control analógico de los valores de salida del problema a resolver por la comunidad de bacterias como forma de facilitar la comprensión de los resultados.

La metodología propuesta se aplica a la resolución de una sencilla instancia del problema TSP (*Traveling Salesman Problem*) con tres ciudades y tres aristas que los unen, para lo cual se hace uso de información lumínica como medio de evaluación de datos. Siguiendo el algoritmo que se detallará más adelante en este capítulo, las soluciones a este TSP no se evaluarán por el color de fluorescencia que expresen, sino por la intensidad de ésta.

Como paso previo a la evaluación de las soluciones, emulando los pasos de un

algoritmo genético, han de formarse éstas. Es aquí donde interviene el complejo de recombinación *Hin/hix* que, como luego se especificará, ayuda a explorar el espacio de búsqueda del problema.

Es importante hacer hincapié en dos características ventajosas del modelo propuesto:

- Autonomía del biodispositivo. Desde la codificación genética inicial de los datos del problema hasta la obtención de un resultado, todo el proceso de computación se realiza de manera autónoma en la comunidad de bacterias. Es decir, la comunidad no necesita señalización externa para la correcta consecución del problema.
- El proceso de exploración del espacio de búsqueda, donde se forman posibles soluciones al problema, y el de evaluación de éstas de manera analógica se realizan de forma simultánea. El algoritmo permite la evaluación de las soluciones durante el mismo proceso de formación.

5.1. Complejo *Hin/hix*: antecedentes

Este mecanismo constituye una potente herramienta para realizar recombinaciones genéticas aleatorias y se fundamenta en las proteínas *Hin* y los sitios de recombinación *hix* que presenta la bacteria *Salmonella*. En ésta bacteria, el complejo *Hin/hix* es utilizado para llevar a cabo procesos de recombinación específica entre los sitios de recombinación *hixL*, *hixR* y *hixC* con la finalidad de alternar la expresión de sus dos flagelos [Hug88].

Se destacan dos ventajas principales de este mecanismo *in-vivo* que lo convierten en una técnica adecuada para los desarrollos computacionales en bacterias:

- Permite la codificación de los datos del problema en unidades genéticas cuyas posiciones sean intercambiables. De esta manera se autoriza la alteración de la cadena de genes, y se modifica así el orden de los datos. Este proceder resulta de gran importancia para aquellos problemas que basen sus soluciones

en la ordenación circunstancial de los datos iniciales, sirviendo aquí el complejo *Hin/hix* como método de exploración del espacio de búsqueda del problema a resolver.

- Debido a la maleabilidad de sus componentes, el complejo *Hin/hix* puede emplearse de distintas formas gracias a que presenta una amplio abanico de variaciones.

Recientemente han sido diseñados dos computadores bacterianos que, basándose en la utilización del mecanismo descrito anteriormente, resuelven dos problemas matemáticos clásicos: el *Burnt Pancake Problem* (BPP) y el *Hamiltonian Path Problem* (HPP). Ambos computadores se describen en los dos apartados que se encuentran a continuación.

5.1.1. Resolución del BPP

Este típico problema matemático se especifica de la siguiente forma:

- Entrada al problema: un conjunto de porciones de bizcocho. Todas las porciones tienen una característica común consistente en tener una cara quemada y otra sin quemar. Además de esta característica común, tienen otra faceta particular de cada porción: la longitud. Cabe destacar que todas las porciones tienen una longitud distinta.
- Objetivo: la colocación de las porciones apilándolas una sobre otra. La pila final debe estar ordenada, desde la porción más grande (abajo) a la porción más pequeña (arriba), con todas las porciones mostrando su cara sin quemar en la parte superior.

Los autores del experimento resolvieron *in-vivo* un BPP sencillo con una entrada consistente en dos porciones [Hay08]. Para ello codificaron los datos de entrada (cada porción) en fragmentos de ADN con funcionalidades específicas y usaron el

complejo Hin/*hix* para intercambiar su orden y orientación. Debido a la funcionalidad de cada cadena de ADN, sólo el orden correcto de las porciones produce una respuesta genética concreta, identificando así la salida de la computación.

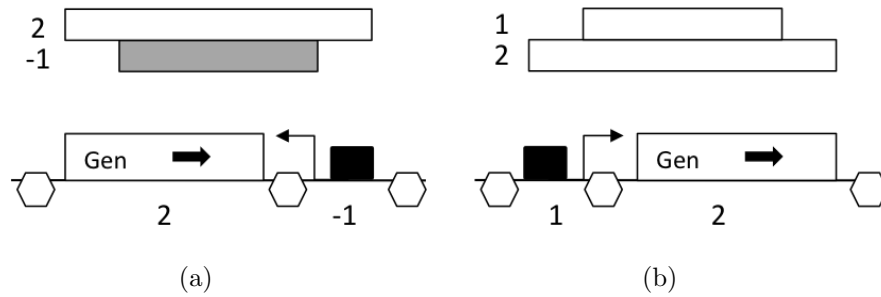


Figura 5.1: Dos posibles configuraciones resultado de la actividad de la proteína Hin sobre los sitios de recombinación *hix* (representados por hexágonos). a) Configuración errónea por dos motivos: orden de las porciones (la mayor se sitúa sobre la menor) y la segunda porción está invertida (el color gris simboliza la parte quemada). Como consecuencia, el gen no se expresa. b) Colocación correcta de los módulos de ADN que concluye en la expresión del gen estructural.

En la figura 5.1 puede observarse la correspondencia entre la abstracción del problema y su codificación genética equivalente.

De las dos porciones que conforman la entrada de este ejemplo, a la menor se la representa mediante un promotor¹ y al mayor se le hace corresponder un gen estructural con información de resistencia antibiótica². Ambas regiones se encuentran flanqueadas por sitios de recombinación específica *hixC* que permiten su inversión e intercambio³.

Para formalizar las posibles soluciones, se indica con el número 1 la posición del promotor que habilita su actividad hacia la derecha de la imagen, con el número -1 su posición inversa, con el número 2 la posición del gen con la flecha hacia la

¹Concretamente, el promotor Lac (pLac).

²Se usa un gen resistente a la tetracilina.

³Todos los módulos rodeados por sitios *hixC* son susceptibles de inversión e intercambio por la acción de la proteína Hin.

derecha y con el número -2 la posición inversa del gen. De esta manera, las ocho posibles formas de combinar las dos porciones son:

$$(1, 2), (1, -2), (-1, 2), (-1, -2), (2, 1), (2, -1), (-2, 1), (-2, -1) \quad (5.1)$$

Las soluciones que se muestran en la figura 5.1 corresponden a los casos $(2, -1)$ (5.1a) y $(1, 2)$ (5.1b) y sólo la segunda representa la solución correcta al problema. Esto es debido a que los módulos de ADN presentan una dirección específica ($5'$ a $3'$) y requieren una ordenación concreta para su funcionamiento. Así, el promotor debe inducir la expresión de un gen estructural que se encuentre dirigido hacia el mismo sentido que él (siendo igual de válida la solución inversa $(-2, -1)$).

Al ser un algoritmo desarrollado por células vivas en una población, la computación es totalmente autónoma y al cabo de un determinado tiempo de cultivo, las salidas del problema estarán disponibles para ser leídas.

El hecho de codificar una de las porciones del bizcocho (la más larga) en un gen con información de resistencia a antibióticos, facilita la lectura de los resultados. Así, aquellas bacterias que contengan la solución $(1, 2)$ o la $(-2, -1)$ expresan dicha resistencia por lo que la introducción en la población del antibiótico correspondiente elimina a todas las demás posibles soluciones.

Es importante notar que la salida a este experimento se está midiendo de un modo estrictamente digital con dos estados: bacteria resistente y bacteria no resistente. Con el motivo de ampliar la variedad en la salida de la computación, el desarrollo propuesto más adelante (5.2) proporciona resultados analógicos situados en gradiente.

5.1.2. Resolución del HPP

Resolviendo el problema del camino Hamiltoniano (HPP) se pretende responder si hay o no una ruta que recorra todos los nodos de un grafo dirigido desde uno inicial a otro final pasando una única vez por cada uno.

Los autores de este experimento [Bau09] construyen un computador bacteriano similar al definido en el apartado anterior en el que cada bacteria tiene la capacidad

de evaluar las posibles soluciones del problema planteado y explorar el espacio de búsqueda gracias nuevamente al complejo Hin/hix .

A diferencia del experimento anterior, la salida aquí se da en forma de información lumínica en lugar de resistencia antibiótica (lo que continua siendo una salida booleana). De esta manera, aquella bacteria de la población que encuentre una solución al problema, expresará luz amarilla siguiendo la metodología expuesta a continuación.

La instancia del problema que se pretende resolver consta de un grafo dirigido cuya especificación es la siguiente:

- Nodos. El grafo consta de 3 nodos:
 - ▷ RFP: Nodo inicial, llamado así por expresar luz fluorescente roja (*Red Fluorescent Protein*).
 - ▷ GFP: Nodo intermedio, llamado así por expresar luz fluorescente verde (*Green Fluorescent Protein*).
 - ▷ STOP: Nodo final. Este nodo terminal no codifica ninguna información genética relevante.
- Conexiones: El grafo consta de 3 conexiones:
 - ▷ Arista A: $\{RFP \rightarrow GFP\}$
 - ▷ Arista B: $\{RFP \rightarrow STOP\}$
 - ▷ Arista C: $\{GFP \rightarrow STOP\}$

Este problema tiene un único camino Hamiltoniano posible. Éste comienza en el nodo inicial (RFP), viaja a través de la arista A hasta GFP y termina recorriendo la arista B para acabar en el nodo final.

La figura 5.2 muestra el esquema de codificación del problema en un segmento de ADN. El comienzo de este segmento está formado por un promotor seguido de la primera mitad⁴ del nodo RFP. Esta estructura inicial es siempre fija ya que todos

⁴Recordar que los segmentos de ADN son módulos siempre dirigidos en sentido 5'-3'.

5.2. Definición del algoritmo celular propuesto

En esta sección se presenta el algoritmo que, utilizando un computador bacteriano, debe resolver una instancia del problema del viajante (TSP⁵) utilizando como herramienta principal la recombinación aleatoria proporcionada por el complejo Hin/hix .

El problema TSP es un problema NP-Completo en el que, dado un grafo no dirigido $G = \{N, A\}$, donde N es el conjunto de nodos en el grafo y A el conjunto de aristas que los une, hay que encontrar el camino más corto posible que visite cada nodo exactamente una vez partiendo de un determinado nodo inicial. Para ello, cada arista del conjunto A está formada por una tupla $[C_o, C_d, D]$ en la que C_o representa la ciudad origen, C_d la ciudad destino y D la distancia específica de esa arista.

La complejidad extra que acarrea el TSP radica en su salida. Así, se ha visto como el HPP presentaba una salida completamente booleana al encontrar o no un camino Hamiltoniano en el grafo. Dada una posible solución del HPP es posible evaluarla sin tener en cuenta el resto del espacio de búsqueda del problema. Es decir, basta con comprobar las condiciones que ha de cumplir un camino Hamiltoniano para saber si esa posible solución es válida o hay que continuar el proceso de computación.

De modo similar el problema BPP ha de dar como salida la única ordenación correcta de todas las porciones de bizcocho, no siendo válidas las demás configuraciones que se puedan formar durante el problema. El punto en común entre este problema y el HPP se encuentra en la forma de evaluar las posibles soluciones. Para ambos, la información que aporta un sólo individuo del espacio de búsqueda es suficiente para: 1) dar una respuesta positiva al problema, ó 2) continuar el proceso de búsqueda de la solución.

Sin embargo, al ser la salida del TSP el camino más corto de todos los posibles, el proceso de evaluación de las posibles soluciones ha de contar con el resto del espacio de búsqueda ya evaluado. Esto es así ya que se debe comparar la distancia

⁵Del inglés *Traveling Salesman Problem*.

del individuo que está siendo evaluado con la distancia de los demás.

Es por ello que una salida estrictamente digital que se base en la validez o no (expresión lumínica o no) de una determinada solución local no es adecuada para este desarrollo. El algoritmo propuesto en este apartado produce, en lugar de una señal lumínica digital, una expresión de los genes de fluorescencia gradual basada en su intensidad [Hei95] [Sha05]. Es decir, la intensidad lumínica que muestre una bacteria dependerá de la distancia del camino o individuo del espacio de búsqueda que haya formado.

De esta forma, la bacteria que contenga la solución del problema será fácilmente detectada gracias a la salida analógica en gradiente que presenta la población.

5.2.1. Diseño del biodispositivo

La instancia del TSP para la cual el algoritmo da respuesta, consta de un grafo no dirigido cuya especificación es la siguiente:

- Nodos: X -nodo inicial-, Y y Z .
- Conexiones: A ser un grafo no dirigido, todos los nodos se encuentran conectados entre sí. Las distancias se especifican de la siguiente manera:
 - ▷ Conexión $X - Y$ ó Camino 1: Distancia larga.
 - ▷ Conexión $X - Z$ ó Camino 2: Distancia media.
 - ▷ Conexión $Y - Z$ ó Camino 3: Distancia corta.

Ya que la solución al problema debe ser presentada en forma de secuencia de caminos, cada uno se corresponderá con un gen de fluorescencia cuya expresión lumínica varíe en intensidad dependiendo de la longitud de éste. De esta manera, las posibles soluciones al problema planteado son secuencias de genes con diferentes intensidades que serán sumadas en el interior de la bacteria para expresar la fluorescencia correspondiente a la longitud total de los caminos que forman un individuo concreto.

Para la implementación *in-vivo* de este diseño (al igual que los anteriores experimentos) todas las células que pertenezcan a la población bacteriana proceden de una misma cepa modificada transgénicamente y por tanto, todas las bacterias presentan el mismo conjunto de funcionalidades.

Para construir un biodispositivo complejo en el que haya bacterias con características y roles diferenciados hace falta diseñar también las comunicaciones entre ellas con el fin de sincronizar sus funciones computacionales. La tercera parte de la presente Tesis Doctoral tratará sobre éste último aspecto de la computación bacteriana.

La utilización del mecanismo *Hin/hix* confiere a la comunidad una característica única entre los biodispositivos uni-cepa: la capacidad de comportamiento aleatorio.

Esto es así ya que a diferencia de una población en la que cada bacteria responda de la misma manera a un estímulo sin haber lugar a comportamientos alternativos, la acción de la proteína *Hin* sobre los sitios de recombinación específica *hix* da lugar a fragmentos funcionales de ADN distintos en cada célula.

Es por ello que este paradigma de cómputo puede ser visto como el paso inmediatamente anterior a la computación con comunidades de bacterias multi-cepa.

El segmento de ADN en el que se codifican los datos iniciales del problema se especifica en la figura 5.3. Siendo éste un módulo similar a los codificados en experimentos anteriores, una importante ventaja (aparte del comportamiento analógico de la salida) radicaría en la localización de éste.

Hoy en día, la investigación centrada en el genoma bacteriano y su secuenciación han revelado la existencia de los denominados *neutral sites* o sitios neutrales en el genoma de las bacterias, de forma que la inserción de módulos funcionales en estas partes del genoma no afectan en absoluto las funcionalidades de la célula y su viabilidad. Son estos sitios buena alternativa a los plásmidos como portadores de la información del problema, evitando así muchos inconvenientes derivados del uso de éstos tales como procesos de conjugación incontrolados o la necesidad de usar antibióticos.

Referente a la codificación del segmento 5.3 y cumpliendo los requisitos iniciales del problema, se considera que:

- Conjunto $RBS1 + \text{gen } Camino1$: expresión lumínica alta.
- Conjunto $RBS2 + \text{gen } Camino2$: expresión lumínica media.
- Conjunto $RBS3 + \text{gen } Camino3$: expresión lumínica baja.

Es importante notar que en el segmento de codificación hay sólo dos promotores de manera que dos únicos genes (todos los caminos del problema están compuestos por dos genes) pueden ser expresados a la vez en una bacteria concreta.

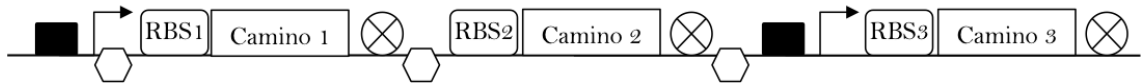


Figura 5.3: Fragmento de ADN correspondiente a la codificación de los datos del problema propuesto. La secuencia de *stop* se denota por \otimes . *RBS*: *Ribosome Binding Site*. Los módulos formados por $RBS1+Camino1+\otimes$ y $RBS2+Camino2+\otimes$ están rodeados por sitios de recombinación *hix*.

Ya que las posibles soluciones a este TSP de tres nodos deben acabar por recorrer el *Camino3*, la posición en que se coloca éste en el segmento de codificación debe permanecer inalterada siendo este gen expresado en cualquier solución formada. No es así en el caso del resto de caminos, los cuales no pueden formar parte simultáneamente de la misma solución.

Por consiguiente, los sitios de recombinación específica *hix* son situados rodeando los genes *Camino1* y *Camino2*. La finalidad es permitir a cada bacteria el intercambio aleatorio entre ambos.

Como resultado a este proceso, el par de genes $\{Camino1, Camino3\}$ o el par $\{Camino2, Camino3\}$ será expresado en cada célula de la población. Así, se consigue que empleando una única señal de salida como es la fluorescencia de un color específico, el significado de ésta se aumente mediante la graduación en intensidad de la solución.

5.2.2. Simulación computacional

En este apartado se lleva a cabo la simulación computacional del biodispositivo diseñado anteriormente con el fin de mostrar gráficamente el comportamiento de la comunidad.

Es importante aquí hacer un estudio de las ecuaciones utilizadas en la simulación de la expresión genética ya que son las mismas que guían las simulaciones de los posteriores desarrollos de esta Tesis Doctoral. Están basadas en las ecuaciones diferenciales de Michaelis-Menten⁶ las cuales se detallan a continuación así como su traducción al problema que se pretende simular.

La ecuación tipo para representar la transcripción genética por parte de un promotor de un determinado gen estructural está marcada por la tasa de producción menos la de degradación, siendo esta última una tasa estrictamente lineal.

Esta ecuación se define de la forma siguiente:

$$\mu(t) = p(t) - \delta\mu(t) \quad (5.2)$$

donde $\mu(t)$ representa la expresión genética final, $p(t)$ la tasa de producción y $\delta\mu(t)$ la tasa lineal de degradación.

La tasa de producción depende del tipo de promotor (inducible, represible o constitutivo), las características del factor de transcripción (FT) (simple, múltiple, nivel de transcripción, etc...) y la actividad del promotor, entre otras cosas.

El modelo que representa la tasa de producción de un promotor inducible, por el cual el FT incrementa la tasa de transcripción, se define como:

$$p(t) = \beta \frac{FT}{\gamma + FT} + \alpha \quad (5.3)$$

donde β identifica la tasa máxima de transcripción, γ representa una constante de saturación y α una constante de transcripción por defecto (la cual, por simplificación, se considerará nula en posteriores desarrollos).

⁶www.biochemweb.org

El modelo correspondiente a la acción de un promotor represible por acción externa es:

$$p(t) = \beta \frac{1}{\gamma + FT} + \alpha \quad (5.4)$$

donde las variables presentan el mismo significado que en la ecuación 5.3. Ahora, el FT disminuye la tasa de transcripción.

Sustituyendo el valor de la variable $p(t)$ en la ecuación 5.2 y considerando un factor de transcripción FT variable en el tiempo, se obtiene para el caso de un promotor inducible la siguiente ecuación de transcripción genética:

$$\mu(t) = \beta \frac{FT(t)}{\gamma + FT(t)} - \delta \mu(t) \quad (5.5)$$

Si FT no varía en el tiempo se obtiene, en el caso de un promotor inducible, la siguiente ecuación de transcripción genética:

$$\mu(t) = \beta \frac{FT}{(\gamma + FT)\delta} \quad (5.6)$$

la cual alcanza un estado estable.

Por último, cabe destacar la acción del proceso de translación para formar la proteína final. Este proceso añade una degradación a la producción de proteínas, la cual es proporcional a la cantidad de mRNA resultado de la transcripción genética $\mu(t)$. Así, esta ecuación se define de la forma:

$$\eta(t) = \lambda \mu(t) - \delta_p \eta(t) \quad (5.7)$$

donde λ es una constante y δ_p el factor de degradación de la producción proteica $\eta(t)$.

Siendo a día de hoy imposible simular todos los procesos metabólicos internos de una célula para obtener resultados 100 % fiables, se presenta aquí la simulación del comportamiento de los mecanismos básicos involucrados en la expresión genética tales como promotores, RBS, mRNA o genes [Kid07], con el objetivo de emular el uso de biobricks⁷⁸.

⁷Harvard MIT and UCSF joint project. <http://www.biobricks.org/>

⁸Se pueden encontrar modelos matemáticos actualizados en la página web del proyecto OpenWetWare. <http://www.openwetware.org>

Se definen así las siguiente ecuaciones:

$$PoPs_{out} = K_t \quad (5.8)$$

$$PoPs_{out} = \frac{maxK_r [activator]}{K_d + [activator]} \quad (5.9)$$

$$PoPs_{out} = \frac{maxK_t}{K_d + [activator]} \quad (5.10)$$

$$\frac{d[mRNA]}{dt} = PoPs_{in} - K_{deg} [mRNA] \quad (5.11)$$

$$SynthesisRate = K_{tr} [mRNA] \quad (5.12)$$

$$\frac{d[protein]}{dt} = SynthesisRate - K_{deg} [protein] \quad (5.13)$$

cuya explicación se detalla a continuación:

- La ecuación 5.8 es una ecuación lineal y constante que representa la actividad de un promotor constitutivo.
- Las ecuaciones 5.9 y 5.10 corresponden a los modelos de Michaelis-Menten 5.3 y 5.4 de promotores inducibles y represibles respectivamente. Para ello, se considera la variable α como nula.
- La ecuación 5.11 resta la degradación en la transcripción $\delta\mu(t)$ de la ecuación 5.2 mediante la constante K_{deg} . Se simula así la actividad del mRNA.
- La ecuación 5.12 representa la acción de la tasa de transcripción β , siendo un símil de la actividad del RBS (*Ribosome Binding Site*).
- La ecuación 5.13 simula la acción del factor de degradación $\delta_p\eta(t)$ en la producción de proteínas de la ecuación 5.7 mediante la constante K_{deg} .

Muchas de las ecuaciones anteriores son no lineales representando así el movimiento, evolución y comportamiento de los procesos celulares. Sin embargo, y debido a que el motivo de simular bloques de biobricks es poder construir y diseñar sistemas genéticos complejos utilizando un conjunto reducido de bloques, las aproximaciones matemáticas de las ecuaciones lineales permitirían un cálculo más sencillo de los resultados.

Un modelo basado en biobricks simulados con ecuaciones diferenciales ordinarias (ODE's) tendría un número alto de parámetros locales con el fin de especificar la tasa de cambio del sistema. Cuanto mayor es el circuito a diseñar mayor es también el número de parámetros a definir lo cual aumentaría la tasa de errores y la dificultad en el diseño.

Por ello se realizan procesos de linealización sobre aquellas ecuaciones no lineales que se necesitan para la construcción de los bloques [Kid07].

Se llega así a obtener la ecuación siguiente:

$$\frac{mRNA[out]}{PoPs[in]} = \frac{1}{s + K_{deg}} \quad (5.14)$$

resultado de linealizar la ecuación 5.11

Igualmente se obtiene:

$$\frac{Protein[out]}{SynthesisRate[in]} = \frac{1}{s + K_{deg}} \quad (5.15)$$

como resultado de la linealización de la ecuación diferencial 5.13.

En las simulaciones del biocircuito y el oscilador propuestos en la tercera parte de esta Tesis se explica con más detalle el uso de los biobricks simulados construidos con estas ecuaciones. Por tanto, aquí se especifican únicamente los resultados finales de la simulación de la comunidad bacteriana modificada para resolver el TSP de 3 nodos.

La figura 5.4 muestra los resultados de la simulación donde dos gráficos diferentes, correspondientes a la expresión de cada camino (gráfica 5.4a) y al comportamiento global de la comunidad (gráfica 5.4b), se miden en términos de tiempo e intensidad.

En la gráfica 5.4a se observan tres funciones de salida cuya intensidad es directamente proporcional a la longitud de los caminos mientras que la gráfica 5.4b muestra la fluorescencia correspondiente a la formación de las dos rutas solución posibles al TSP propuesto. La aparición de estas dos últimas rutas significa que dos subgrupos bacterianos pueden ser visualmente distinguidos en la población inicial, donde cada uno expresa una diferente intensidad lumínica correspondiendo al par $\{Camino1, Camino3\}$ o $\{Camino2, Camino3\}$ que se encuentra codificado en cada una de sus bacterias.

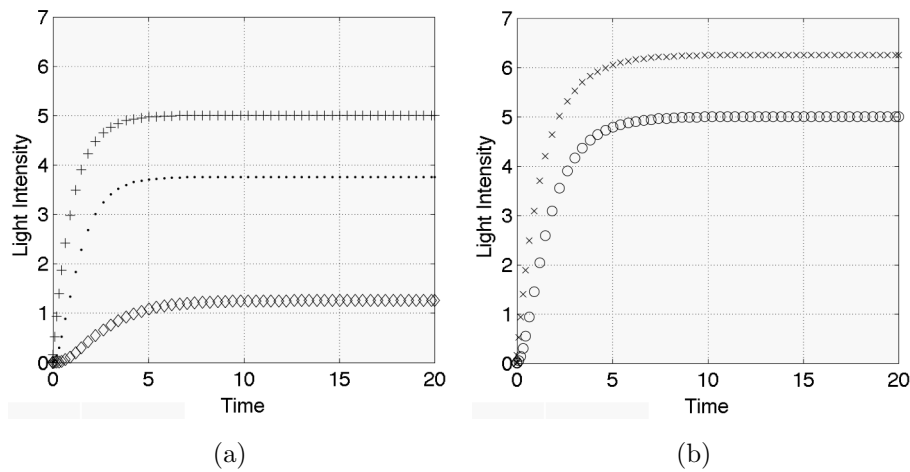


Figura 5.4: Resultados de la simulación del TSP. a) Gráfica que muestra la expresión de los tres caminos por separado; *Camino1* (+); *Camino2* (·); *Camino3* (◊). b) Gráfica que muestra la expresión lumínica de los dos posibles tipos de bacterias formadas por el algoritmo: las que contienen el recorrido $\{Camino1, Camino3\}$ (×) y las que contienen codificado el recorrido $\{Camino2, Camino3\}$ (◊).

Cabe destacar que, aparte de los dos niveles de fluorescencia mostrados en la 5.4b aparecerían en la población bacterias con otro tipo de fluorescencia: la correspondiente al *Camino3*. Esto es debido al funcionamiento del complejo *Hin/hix* ya que, en ocasiones, además de cambiar de lugar los genes, éstos se giran (según el eje transversal) dando como resultado un segmento de ADN sobre el que no funciona la actividad del promotor.

La característica de inversión que presta el complejo *Hin/hix* es cada día más

maleable y puede llegar a ser prácticamente inapreciable. Su presencia en la simulación dependerá del valor probabilístico que se le atribuya.

Parte III

Arquitecturas de comunicaciones

Capítulo 6

Preliminares

Índice

6.1. Procesamiento paralelo	48
6.2. Paralelismo natural	53

En capítulos anteriores se aborda la computación celular considerando la bacteria como objeto último de cómputo. Es decir, los desarrollos vistos hasta ahora se diseñan de forma que el problema completo (de principio a fin) se resuelva en el interior de la célula. Así en una población de bacterias habrá tantas copias del computador celular como células y es la estructura interna de una bacteria aislada lo único importante.

Sin embargo, la naturaleza *emplea* las bacterias de una forma muy distinta agrupándolas en sociedades llamadas biopelículas [Cos95] [Cos99] (*biofilms* en inglés) en las que cientos de distintas especies bacterianas forman una comunidad estable y robusta mediante sus habilidades de coordinación y comunicación. El objetivo de poder alterar y modificar estas comunicaciones en un laboratorio para servir a fines específicos representa un importante reto en la ciencia que todavía está en una fase muy inicial [Bal10].

Así, esta tercera parte propone tres diseños de comunidades heterogéneas¹ de

¹Se entiende aquí por comunidad heterogénea a aquella comunidad con más de una cepa bac-

bacterias que persiguen un objetivo común.

El control y manipulación de la comunicación y sincronización extracelular e intercelular es la base en la que se fundamentan estos diseños mediante protocolos o arquitecturas de comunicaciones específicos aplicados a los problemas que se quieran resolver. Más allá de explotar las funcionalidades de comunicación entre bacterias de la misma cepa, estos desarrollos exploran las posibilidades que la mezcla de cepas bacterianas en un mismo cultivo pueden tener para la formación de sociedades que emulen aquellas formadas naturalmente.

Es importante notar que no se emplean distintas especies en el diseño de los biodispositivos sino varias cepas transgénicamente modificadas de la misma especie de bacteria. Debido a la amplia gama de posibilidades de manipular las funcionalidades de una célula², lo cual lleva a formar cepas muy diferentes entre sí, se podría introducir el término de *especies artificiales* ya que las cepas no son de distintas especies pero tampoco presentan los suficientes rasgos comunes para considerarlas como tal.

Partiendo de dos protocolos de comunicaciones de propósito general usados por las bacterias en la naturaleza, la transferencia horizontal de información genética y el *quorum sensing*, se especifican tres arquitecturas cuyas aplicaciones son: resolución de una instancia del problema SAT, construcción de un circuito lógico XOR y el diseño de una señal síncrona de oscilación.

Al igual que los investigadores e inventores de las Ciencias de la Computación basaron sus desarrollos en una sola unidad de proceso antes de expandir el objetivo a la computación paralela, la Biología Sintética se centra durante su primera fase, en la creación de computadores o unidades de proceso unicelulares. Así, resulta conveniente echar un vistazo de manera poco profunda a los principales modelos de paralelización de procesadores para que éstos sirvan de inspiración en los siguientes modelos de computadores bacterianos.

teriana.

²Incluyendo la síntesis de células mínimas, como se verá más adelante.

6.1. Procesamiento paralelo

La computación paralela es un método por el que se dividen grandes problemas en tareas o procesos más pequeños y surge como una respuesta de la arquitectura de computadores a las crecientes demandas de potencia de cómputo de los usuarios. Hasta hace poco tiempo la mayoría de los ordenadores eran uniprosesadores (cuentan con un solo procesador) y esto quiere decir que realizan procesamiento secuencial y únicamente pueden llevar a cabo una acción en un momento determinado.

Esto se contrarresta con la aparición de arquitecturas de computadores complejas que implementan ciertos servicios de concurrencia dando la impresión de paralelismo. Estas soluciones son realmente eficaces, ya que mejoran el rendimiento del ordenador, pero no dejan de ser soluciones secuenciales.

Desde hace más de 30 años, investigadores de todo el mundo han centrado su atención en el procesamiento paralelo. La posibilidad de realizar no una sino más acciones en un determinado momento es, en verdad, muy atractiva.

La idea básica que hay detrás del procesamiento paralelo es que varios dispositivos de proceso (procesadores), ejecutando simultánea y coordinadamente las tareas, puedan rendir más y mejor que un único dispositivo de las mismas características. Se trata ahora, por tanto, de buscar qué innovaciones tecnológicas son necesarias para hacer posible este desarrollo. El concepto de procesamiento paralelo que se ha explicado se observa en la figura 6.1, que ilustra el objetivo de la paralelización como la descomposición en subtareas de una tarea mayor con el fin de resolverla en un periodo de tiempo menor.

Existen muchas maneras de diseñar una arquitectura paralela. Antes de pasar a definir las principales arquitecturas hay que familiarizarse con un término: la escalabilidad. La escalabilidad se define como la capacidad de un sistema (hardware/software) paralelo en ampliar el número de procesadores manteniendo un incremento de la eficiencia.

La escalabilidad depende fundamentalmente de los recursos que los procesado-

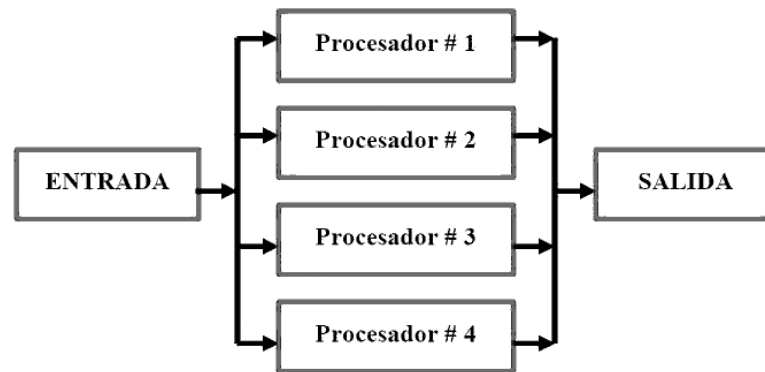


Figura 6.1: Concepto del procesamiento paralelo

res compartan y de la aplicación que va a usar la arquitectura paralela. Obviamente, dependiendo del modelo de funcionamiento de la aplicación habrá que pensar en un modo u otro de arquitectura. Los recursos compartidos se pueden ver sobrecargados si son muchos los procesadores que intentan acceder a ellos. Con todo ello se aprecia que no resulta nada fácil la tarea de aumentar la escalabilidad de una arquitectura paralela.

Aunque la presente Tesis Doctoral no tenga el objetivo de proponer nuevos modelos de paralelización de procesadores ni estudiar en detalle éstos, sí es conveniente mostrar los conceptos básicos de la paralelización digital como antecedentes de la paralelización biológica que se presenta en esta tercera parte. Se pueden mencionar tres tipos principales de arquitecturas paralelas que siguen los ordenadores eléctricos actualmente: multiprocesamiento simétrico, procesamiento masivamente paralelo y procesamiento escalar.

A la vista de los tres modelos explicados a continuación, es notable la inspiración que guía el desarrollo de los diseños celulares de capítulos posteriores.

6.1.1. Multiprocesamiento simétrico

Éste es un modelo de multiprocesador el cual está dotado de un único sistema operativo que proporciona interacción entre los procesadores y sus programas a nivel de procesos y datos. Tiene un solo espacio de memoria compartido por lo que

es llamado un sistema fuertemente acoplado (como se ve en la figura 6.2). El hecho de que tenga una sola memoria global hace que el problema de la escalabilidad sea realmente fuerte en este tipo de arquitectura ya que son muchos los recursos a compartir. En multiprocesadores de este tipo el rendimiento del sistema suele subir hasta los nueve procesadores conectados. Por encima de nueve procesadores, el rendimiento cae. Como ventaja cabe destacar la facilidad de programación.

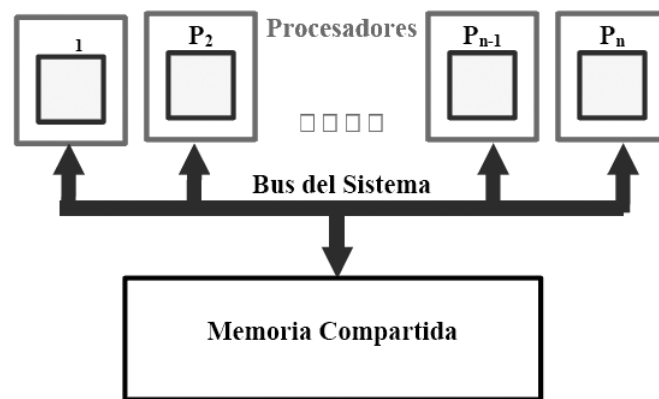


Figura 6.2: Arquitectura de multiprocesamiento simétrico

Son varias las similitudes que se encuentran, desde el punto de vista computacional, entre el esquema de la figura 6.2 y una comunidad de bacterias. Así, cada procesador equivaldría a una bacteria individual y la memoria compartida a la solución nutriente que sirve como sustrato y contenedor a las células. El bus del sistema lo forman los mecanismos que permiten a la célula expulsar al exterior una molécula producida por cascadas metabólicas internas.

Esto es así ya que uno de los principales paradigmas de comunicación celular, el método de *quorum sensing*, basa su funcionamiento en la presencia o ausencia de moléculas específicas producidas por las bacterias en la solución nutriente. Esta funcionaría, por tanto, de memoria compartida en las que las células *escriben* y *leen* estas moléculas para la correcta sincronización de la población. En capítulos posteriores se especificará más en detalle el funcionamiento de esta técnica de comunicación célula-célula.

6.1.2. Procesamiento masivamente paralelo

Éste es un modelo de multicomputadora. Recibe este nombre por estar formado por múltiples computadoras independientes, cada una con su procesador y su memoria. Debido a que no comparte memoria evita los cuellos de botella que se producían en el caso anteriormente estudiado. La información que tengan que intercambiar este estilo de multicomputadoras se realiza mediante paso de mensajes formando así una topología de red como se muestra en la figura 6.3.

Esta arquitectura permite la construcción de sistemas de gran tamaño, con cientos e incluso miles de procesadores, por lo que ésta es una tecnología altamente escalable. A medida que aumente el número de procesadores el rendimiento se incrementará, siendo el gran inconveniente el coste y el espacio físico requerido para la correcta implementación de este modelo.

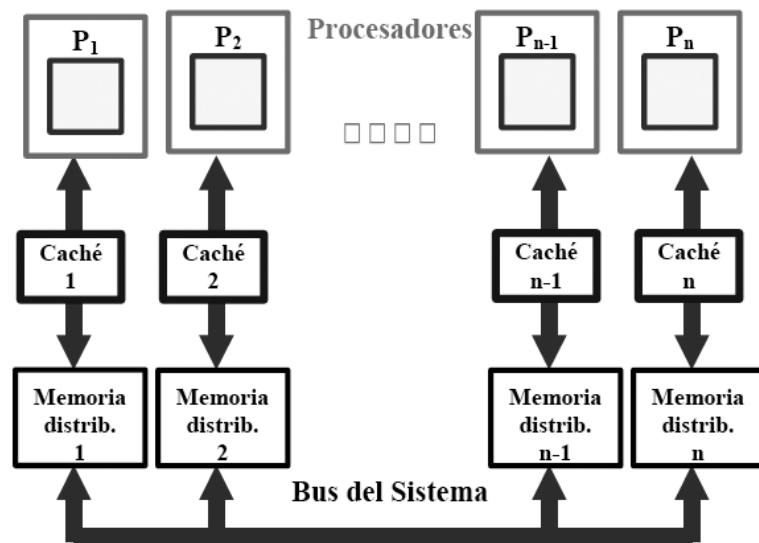


Figura 6.3: Arquitectura del procesamiento masivamente paralelo

Este modelo de paralelización no va a servir de inspiración para los diseños presentados en esta tercera parte debido a la complejidad en la equivalencia de este modelo con una población de bacterias. Sin embargo, sí que pueden extraerse de aquí interesantes conclusiones como la modificación de la fisionomía de la población para permitir el uso de memorias localizadas.

Ya se han desarrollado varios trabajos describiendo diferentes sustratos orgánicos (hidrogel) e inorgánicos (sol-gel) para estabilizar las bacterias [Bje06] así como métodos de láser para situar bacterias en formaciones específicas tales como matrices en distintas superficies [Bar04]. De este modo se puede restringir el área de acción de cada bacteria habilitando los procesos de *lectura* de las moléculas a determinadas bacterias.

6.1.3. Procesamiento escalable

Es un modelo híbrido entre los dos anteriores. Se basa en utilizar dos niveles de memoria, una para uso particular de cada procesador y otra para compartir entre todos (figura 6.4). Esta arquitectura combina las ventajas de las ya vistas anteriormente: es fácilmente programable y de gran escalabilidad. Si con un procesador, esta arquitectura trabajaría al 100 % de su capacidad de procesamiento, con 16 procesadores el rendimiento del computador se vería incrementado a un 1100 %.

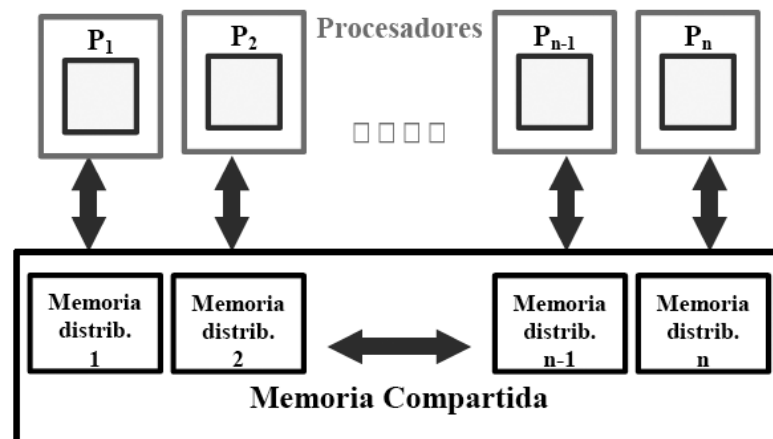


Figura 6.4: Arquitectura de procesamiento escalable

Otra visión de la memoria distribuida, más fiel a la realidad biológica, es considerar ésta como el espacio interior a cada membrana celular. En efecto, los procesos metabólicos que tienen lugar en el interior de una bacteria no son compartidos por las demás, por lo que cada bacteria de la población cuenta con su propia memoria privada.

De esta forma, la figura 6.4 es la que más fielmente refleja la equivalencia con los diseños bacterianos que se presentarán en capítulos posteriores. Por otro lado, el intercambio de información genética que ocurre en el paradigma de transferencia horizontal, necesita del contacto entre células para poder realizarse. Este intercambio se llevaría a cabo desde memoria privada a memoria privada, sin pasar por la memoria común. Sería por tanto necesaria la existencia de un bus similar al del modelo de la figura 6.3 entre las memorias distribuidas. Este bus permitiría los flujos de información genética.

6.2. Paralelismo natural

Hasta hace poco más de 20 años se consideraba a la bacteria como un ser no social e individual, incapaz de mantener comunicación alguna con el exterior. La cualidad de comunicación se creía propiedad de organismos superiores como los humanos. Sin embargo, el descubrimiento de modelos de sociedades bacterianas en las que un fin colectivo movía a las bacterias hacia comportamientos de sincronización entre ellas, resultó un importante descubrimiento. Hoy en día, sigue teniendo gran importancia en el mundo de la investigación el estudio de estas sociedades y las capacidades de colaboración de las bacterias.

El reduccionismo³ que ha guiado los estudios de la biología durante cientos de años hacía imposible el estudio de estos fenómenos sociales ya que el ensamblaje de pequeñas partes no es suficiente para explicar éstos. Por ello, en los últimos años se aborda el problema de las sociedades y biopelículas de bacterias desde un punto de vista holista⁴.

Desde la regulación genética en el interior de la célula a las sociedades de bacterias, la microbiología está experimentando una nueva fase que enfatiza las interacciones de diversos elementos que forman una comunidad. Esta nueva visión

³Enfoque según el cual, la reducción a las partes es suficiente para el entendimiento del todo.

⁴Enfoque según el cual, las propiedades de un sistema no pueden ser explicadas por el detalle de sus partes por sí solo.

está cambiando la comprensión de la fisiología celular y la capacidad de tratar infecciones bacterianas. Desde este punto de vista, se estudian biopelículas de la cavidad bucal [Kur07] que, con más de 700 especies habitando en sociedad, son de una complejidad asombrosa. En ellas, queda claro cómo el todo es mucho más que la simple suma de sus partes y cómo las interacciones entre especies llevan a funcionalidades totalmente incomprensibles mediante el estudio de una especie aislada.

Es comprensible que la potencia de cómputo, en cuanto a variedad de funcionalidades se refiere, es mucho mayor en una sociedad de bacterias formada por varias especies que en una población con una sola especie bacteriana. En capítulos posteriores se hace hincapié en este aspecto, desarrollando modelos de biodispositivos en los que varias cepas interaccionan para conseguir un fin común.

Un claro indicativo del avance en el diseño sintético de biodispositivos multi-cepa es el número de estas usadas para crear sociedades con fines específicos. El conocimiento actual sobre el metabolismo celular y el control que se tiene sobre él, permite la creación de dispositivos con apenas 2, 3, 4 ó 5 cepas bacterianas, mientras que en la naturaleza existen sociedades de más de 700 especies como ya se ha mencionado.

Esto indica que todavía hay un largo camino que recorrer en la intención de modificar las capacidades de comunicación y sincronización de las bacterias para la consecución de objetivos *artificiales*⁵. Los desarrollos en el campo de la Biología Sintética de los últimos años basados en las capacidades de relación de las bacterias representan el comienzo de este camino, siendo los desarrollos de esta Tesis Doctoral un intento de expandir estos comienzos hacia nuevos límites.

Dos puntos clave son necesarios en la investigación en este campo:

1. La profundización en los conocimientos sobre los mecanismos de sincronización bacterianos.

⁵Se emplea aquí el término *objetivos artificiales* para hacer referencia a aquellos objetivos para los cuales la naturaleza no busca respuesta. Por ejemplo, la resolución de un problema matemático no es un objetivo que resuelva ninguna bacteria en estado natural.

2. La formalización de conceptos basados en la computación y la analogía de éstos con los conocimientos adquiridos en el paso previo.

Los estudios sobre las biopelículas de la cavidad oral en la revisión de Kuramitsu [Kur07] extraen conclusiones esquemáticas importantes sobre las funcionalidades características que presentan las especies bacterianas que conforman estas sociedades. Las interacciones de las bacterias se centran en los siguientes aspectos:

- Los procesos de competición entre bacterias por los nutrientes.
- Las interacciones sinérgicas que pueden estimular el crecimiento o la supervivencia de uno o más residentes.
- La producción de un antagonista por uno de los residentes el cual inhibe el crecimiento de otro.
- Neutralización de un factor de virulencia producido por un residente por la acción de otro.
- Interferencia en las cascadas metabólicas de crecimiento de un residente por otro de la sociedad.

De esta manera, una sociedad de bacterias tan amplia como las biopelículas de la cavidad bucal forman una micro-galaxia única en la que todas las acciones enumeradas anteriormente son entendidas como formas de *guerra y paz* entre los residentes o especies bacterianas de la población. En el capítulo referido a la creación de una señal sincrónica de oscilación se realizan simulaciones computacionales que verifiquen el efecto que tienen en el diseño estas interacciones.

Aunque la mayoría de estas relaciones entre células puedan entenderse de una forma negativa, en el sentido en que una especie intente eliminar a las demás en su búsqueda de la supervivencia, esto no es del todo cierto. Si así fuera, la biopelícula acabaría desapareciendo o, al menos, el número de especies que la conforman decrecería rápidamente. Sin embargo esto no ocurre y la sociedad permanece estable de una manera robusta. Esto indica que las relaciones van mucho más allá de la

agresión, comportándose de una manera colaboracionista por el bienestar de la sociedad. Incluye esta interacción la transferencia horizontal de información genética entre especies de la población con la finalidad de intercambiar aquella información útil para la supervivencia.

Aún estando el conocimiento de estas interacciones en un estado muy inicial, el interés del mundo de la investigación es máximo debido al potencial que las aplicaciones desarrolladas en esta dirección pueden llegar a tener en aspectos computacionales, ecológicos o médicos.

Capítulo 7

Análisis biológico de la comunicación social bacteriana

Índice

7.1. Quorum sensing	59
7.2. Transferencia genética horizontal	64
7.3. Comunicaciones <i>Chip-to-Cell</i>	67

Las bacterias no son seres aislados desde el punto de vista social como se pensaba tradicionalmente, sino que presentan sistemas de comunicación complejos [Kel06] con la finalidad de permitir, a la población en la que viven, perseguir un fin global.

Este capítulo aborda las cuestiones básicas sobre bacterias en cuanto a capacidades de comunicación se refiere para, partiendo de estos mecanismos poder diseñar los biodispositivos con fines computacionales que se detallan en capítulos posteriores.

Conviene comenzar familiarizándose con términos referidos a las realidades sociales de organismos superiores, como los animales, que son también aplicados a las relaciones sociales de las bacterias [Col09] [Wes06]:

- Altruismo: comportamiento entre bacterias de la misma especie que contribuye al bienestar ajeno a expensas del propio. Este acto de *solidaridad* con la población bacteriana en la que habita supone un gasto de energía del individuo que lo lleva a cabo. No hay una relación directa entre el bien conseguido y el sacrificio que supone para la célula ya que si bien esta acción no supone una ventaja tal como la supervivencia de la especie, sí que puede acarrear duras consecuencias como la muerte celular programada (*apoptosis*) del individuo. Sin embargo, en una población relativamente grande, son necesarias las acciones altruistas para la correcta funcionalidad de la especie.
- Canibalismo: es un comportamiento social de las bacterias de una población de la misma especie que demuestra la fuerte *conciencia* de conjunto. Los individuos de una población tienen como objetivo máximo la supervivencia de la especie y, mediante el canibalismo, una determinado número de bacterias es capaz de producir unas sustancias letales que acaben con la vida de los demás. Este comportamiento que, a priori parece negativo, resulta de gran utilidad cuando la población no cuenta con los suficientes sustratos nutrientes y utiliza los restos de las bacterias muertas como tales para el desarrollo normal.
- Cooperación: comportamiento de varios grupos bacterianos de diferente especie (o diferentes cepas de la misma especie) mediante el cual, y siguiendo protocolos de comunicación bien especificados, sincronizan su desarrollo para la supervivencia de todos ellos. Se trata de conseguir formar una comunidad heterogénea estable y robusta.
- Competición: comportamiento dentro de un ecosistema en el que dos o más especies bacterianas (o diferentes cepas de la misma especie) rivalizan por algún factor esencial para la supervivencia de ambas. En una situación de competición habrá, lógicamente, un ganador que será el que logre el objetivo por el que rivaliza. El resto de competidores saldrán perjudicados o incluso desaparecidos.

De entre todos los tipos de relación posibles, la cooperación es el compor-

tamiento social más interesante para el buen entendimiento de los desarrollos de la presente Tesis Doctoral. En concreto, dos protocolos de comunicación sirven como punto de partida para crear comunidades heterogéneas en las que las bacterias cooperen para lograr un fin colectivo único: *quorum-sensing* y conjugación bacteriana.

7.1. Quorum sensing

Se llama quorum-sensing [Jay08] a la comunicación mediante el intercambio de señales en forma de moléculas entre varias bacterias de una población. Esta comunicación se utiliza como una forma de sincronización y cooperación en pos del bienestar de la comunidad bacteriana en su totalidad.

Los sistemas de cooperación basados en quorum-sensing fueron observados por primera vez en la especie bacteriana *Vibrio fischeri* en la que se utiliza para controlar la expresión lumínica de una sociedad [Ncl77]. Posteriormente, se han identificado sistemas que utilizan el mismo tipo de comunicación en muchas especies bacterianas diferentes como *Pseudomonas*, *Escherichia* o *Streptococcus*.

Con el fin de detallar el funcionamiento de este mecanismo, es muy ilustrativo el uso que hace de él la especie *Vibrio fischeri* ya que es prototípico de los sistemas de las bacterias Gram negativas. Este circuito básico de quorum sensing basa su funcionamiento en la regulación del par de proteínas LuxR-LuxI que funcionan como sensor y productor, respectivamente, de moléculas señalizadoras. La proteína que funciona como sensor tiene como objetivo inducir la producción lumínica.

Cuando se expresa el gen estructural que codifica la proteína LuxI se forma ésta, la cual es capaz de sintetizar la molécula de señalización N-Acylhomoserine lactone (AHL). Esta molécula tiene como característica especial la capacidad de ser expulsada al exterior por la bacteria que la produce. En la imagen de la figura 7.1 se observa el ciclo de producción de moléculas AHL.

La figura 7.1a ilustra una situación en la que la baja densidad celular en una población bacteriana determinada conduce a bajos niveles de las moléculas de

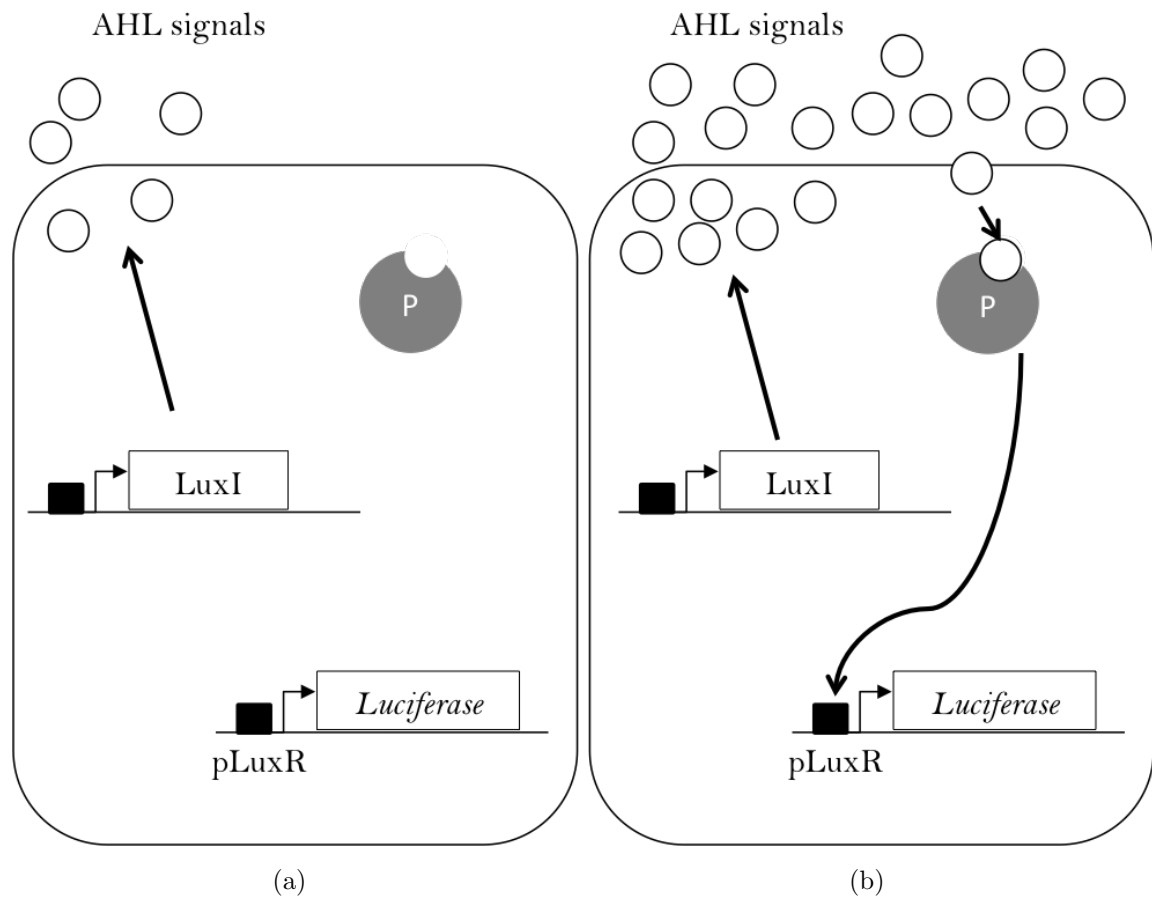


Figura 7.1: Arquitectura básica de un circuito de quorum sensing. *LuxI*: gen estructural cuya expresión da lugar a las proteínas del mismo nombre. Éstas producen las moléculas AHL; P: proteína receptora *LuxR*; *pLuxR*: promotor inducible por la acción de *LuxR*; *Luciferase*: gen objetivo con información genética luminiscente. a) Comienzo del circuito, las moléculas no superan umbral. b) Final del circuito, las moléculas superan umbral expresando el reporter final.

señalización y, consecuentemente, a la baja producción de luminiscencia a través de la expresión de los genes de *Luciferase*. A medida que la densidad celular aumenta, la concentración de las moléculas AHL se hace más presente tanto dentro como fuera de las bacterias.

Cuando la concentración de señales que hay en la solución nutritiva supera un valor umbral específico *programado* en la membrana celular de cada bacteria, las

moléculas AHL penetran en el interior de las células activando las proteínas LuxR. La figura 7.1b representa este comportamiento en el que la proteína LuxR, una vez activada, actúa como factor de transcripción inductor del promotor *pLuxR*. A su vez, este promotor facilita la expresión del gen de Luciferasa que codifica información lumínica.

De esta forma, la población de bacterias se asegura que producirán luz todas al mismo tiempo de una manera síncrona. La comunicación exitosa mediante el mecanismo de quorum sensing proporciona una poderosa herramienta para el diseño de biodispositivos basados en bacterias transgénicas. Esto es, en parte, gracias a la maleabilidad del mecanismo que permite diseñar multitud de pares LuxI-LuxR diferentes para albergar comunicaciones síncronas simultáneas, así como la posibilidad de variar el umbral en uno u otro sentido dependiendo de las funcionalidades deseadas.

Aparte de este sistema básico de quorum sensing, también se pueden observar en distintas especies bacterianas modelos más complejos que siguen arquitecturas de más interés para los desarrollos computacionales. Así, se encuentra una arquitectura paralela en el sistema de quorum sensing de la especie *V. harveyi* y una arquitectura secuencial en la especie *P. aeruginosa*. En las figuras 7.2a y 7.2b, respectivamente, se detallan ambos modelos desde un punto de vista esquemático potenciando la simplicidad en la explicación del mecanismo en detrimento del detalle biológico.

La arquitectura paralela de la bacteria *V. harveyi* basa su funcionamiento en la utilización de la proteína LuxO como intermediario común a través del cual las tres señalizaciones diferentes actúan con el interior de la célula. En un símil computacional podríamos deducir que esta pieza hace las funciones de un *switch* con tres entradas y una sola salida.

Este modelo es muy beneficiario para poblaciones con varios tipos de señalizaciones o sistemas de quorum sensing ya que ahorra medios físicos de lectura de dichas señales agrupándolos todos en una sola proteína. También está actuando el componente LuxO como un filtro que ayuda a sincronizar las señalizaciones y controlar el ruido genético que se pueda producir al coexistir varios sistemas en la misma

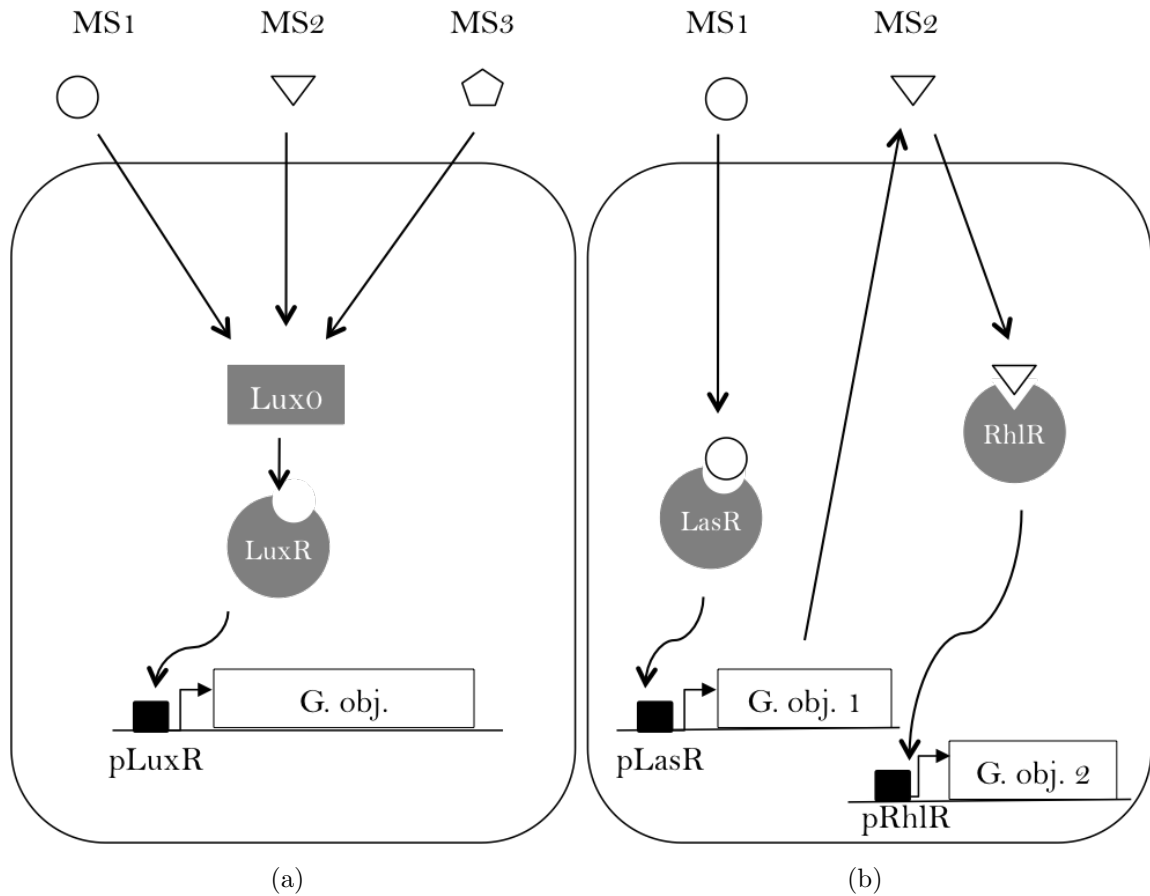


Figura 7.2: Arquitecturas complejas de sistemas de señalización quorum sensing. a) Señalización paralela de la bacteria *V. harveyi*; MS: moléculas de señalización; LuxO: proteína común intermedia; LuxR: proteína receptora; pLuxR: promotor inducible por LuxR; G. obj.: genes objetivo. b) Señalización en serie de la bacteria *P. aeruginosa*; MS: moléculas de señalización; LasR: receptor de MS1; pLasR: promotor inducible por LasR; G. obj. 1: primeros genes objetivo; RhIR: receptor de MS2; pRhIR: promotor inducible por RhIR; G. obj. 2: segundos genes objetivo.

comunidad. Esto hace posible evitar la activación errónea de uno de los sistemas de quorum sensing por la acción de otro.

De modo distinto, los circuitos de quorum sensing de Las y RhI son ejecutados en serie por la bacteria *P. aeruginosa*, donde cada sistema ayuda a regular la expresión genética de un conjunto de genes diferente sincronizados de una manera

secuencial en lugar de paralela (figura 7.2b. Cuando el umbral especificado para el valor de concentración de la molécula de señalización específica del sistema Las (en la figura MS1) es superado, dicha señal se une al receptor LasR activándole. Este, a su vez, induce la expresión genética de los primeros genes objetivo.

Una vez activado el sistema Las, las bacterias comienzan la producción de la segunda molécula de señalización (MS2 en la figura). En el momento en que el valor de la concentración de esta segunda molécula en la solución nutriente es suficiente, tiene lugar la activación del segundo sistema de quorum sensing basado en el circuito Rhl. Así, el receptor RhlR induce la expresión genética del segundo conjunto de genes objetivo.

Por la linea temporal que guía la expresión de ambos conjuntos, se comprende el carácter secuencial del mecanismo de señalización de la bacteria *P. aeruginosa*.

En frecuentes ocasiones, los comportamientos bacterianos descritos al comienzo del presente capítulo (altruismo, canibalismo, competición y cooperación) no se presentan de manera clara en una acción específica de un grupo de bacterias en sociedad. Por ejemplo, muchos de los comportamientos que son regulados por circuitos de quorum sensing, lo cual implica cooperación, tienen como finalidad producir bienes públicos para la sociedad [Wes07]. Tan pronto como esta producción de bienes públicos no se realice por la comunidad en su totalidad, se estaría hablando de un comportamiento altruista, ya que los individuos que participan en la producción de estos factores están gastando una cierta cantidad de energía propia en pos del bien común.

Cuadro 7.1: Bienes públicos

Factor	Rol que desempeña
Invertase	Encima para la digestión de sacarosa
β -lactamase	Inactiva y, por lo tanto, otorga resistencia antibiótica
Biosurfactants	Facilita el movimiento sobre superficies
Microbial repellents	Para repeler competidores

Las bacterias producen de esta forma una gran variedad de elementos, denominados de bien público, que son compartidos con todos los individuos de la población mediante su expulsión al entorno o solución nutriente en la que vive la sociedad. Estando estos bienes públicos codificados en forma de información genética y agrupados en genes estructurales, esta producción es resultado de la inducción en la expresión de los denominados genes objetivo de las figuras 7.1 y 7.2.

Un breve ejemplo de la variedad funcional que los bienes públicos pueden otorgar a la sociedad se muestra en la tabla 7.1.

En los diseños de biodispositivos propuestos en esta tercera parte, se modelan nuevas arquitecturas de quorum sensing afectando más en detalle a las relaciones entre bacterias que a los mecanismos de señalización en sí mismos. Así se emplea siempre una arquitectura algo más compleja que la mostrada en la figura 7.1 ya que se utilizan varias parejas de emisores-receptores de la familia LuxI-LuxR aunque no es tan claro el esquema de la figura 7.2 al que equivaldrían. El objetivo, principalmente del oscilador genético, es diseñar nuevas arquitecturas que afecten al rol que cada bacteria efectúa en la sociedad.

7.2. Transferencia genética horizontal

El otro protocolo de comunicación entre bacterias que se pretende manipular es la transferencia genética horizontal¹ o conjugación bacteriana. Esta técnica confiere a las bacterias, y a otros organismos unicelulares, una característica única entre los seres vivos: la capacidad de compartir información genética entre individuos de la misma generación sin tener que esperar a realizar este traspaso de padres a hijos de manera vertical entre una generación y su generación descendiente.

Esta transferencia horizontal se realiza incluso entre individuos de distinta especie y, a veces, entre distintos taxones en que actualmente se considera subdividida la diversidad de los seres vivos: arqueas (*Archaea*), bacterias (*Bacteria*) y eucariontes (*Eukarya*). La finalidad no es más que la búsqueda de una rápida adaptación al

¹HGT: Horizontal Gene Transfer.

medio a través de un proceso de evolución acelerado [Gog89] [Gog02].

Mediante sencillos intercambios de información genética las bacterias de una población pueden obtener nuevos genes cromosómicos y plásmidos (entre otros) que ya han pasado satisfactoriamente los procesos de selección de otras poblaciones bacterianas de la misma u otra especie. De esta manera, las bacterias pueden ir incorporando a lo largo de su vida nuevas funcionalidades que las permitan adaptarse mejor a su entorno. A su vez, las evoluciones que consigan serán intercambiadas con otros individuos menos desarrollados con el fin de propiciar la supervivencia de todos ellos.

A la transferencia horizontal hay que sumar el papel que en el proceso de evolución desempeña la recombinación² mediante la ordenación aleatoria de genes en búsqueda de nuevas y mejores configuraciones [Lev09].

Esta sección trata de explicar de forma esquemática, y sin detalles biológicos específicos, el mecanismo de conjugación bacteriana. Mediante esta breve explicación el lector podrá estar familiarizado con este mecanismo de comunicación, lo que le facilitará la comprensión de uno de los capítulos posteriores en el que se hace uso de la conjugación para resolver de forma autónoma y evolutiva un problema matemático.

La conjugación bacteriana es, al fin y al cabo, una primitiva forma de sexualidad entre seres vivos. Así, existe una bacteria donante que actuaría en clara analogía con el macho de las especies animales y una bacteria receptora que hace el papel de hembra. Durante el proceso de conjugación, la célula donante dona parte de su material genético a la receptora.

En la figura 7.3 se ilustra el mecanismo de conjugación en cuatro etapas básicas:

- En la figura 7.3a se ve como la bacteria donante y fértil se aproxima a una bacteria receptora infértil. Se denotará la primera bacteria como F+ y la segunda como F-. Es la bacteria donante la que mediante el plásmido que le confiere la característica F+, originará todo el proceso de transferencia horizontal.

²HGR: Homologous Gene Recombination.

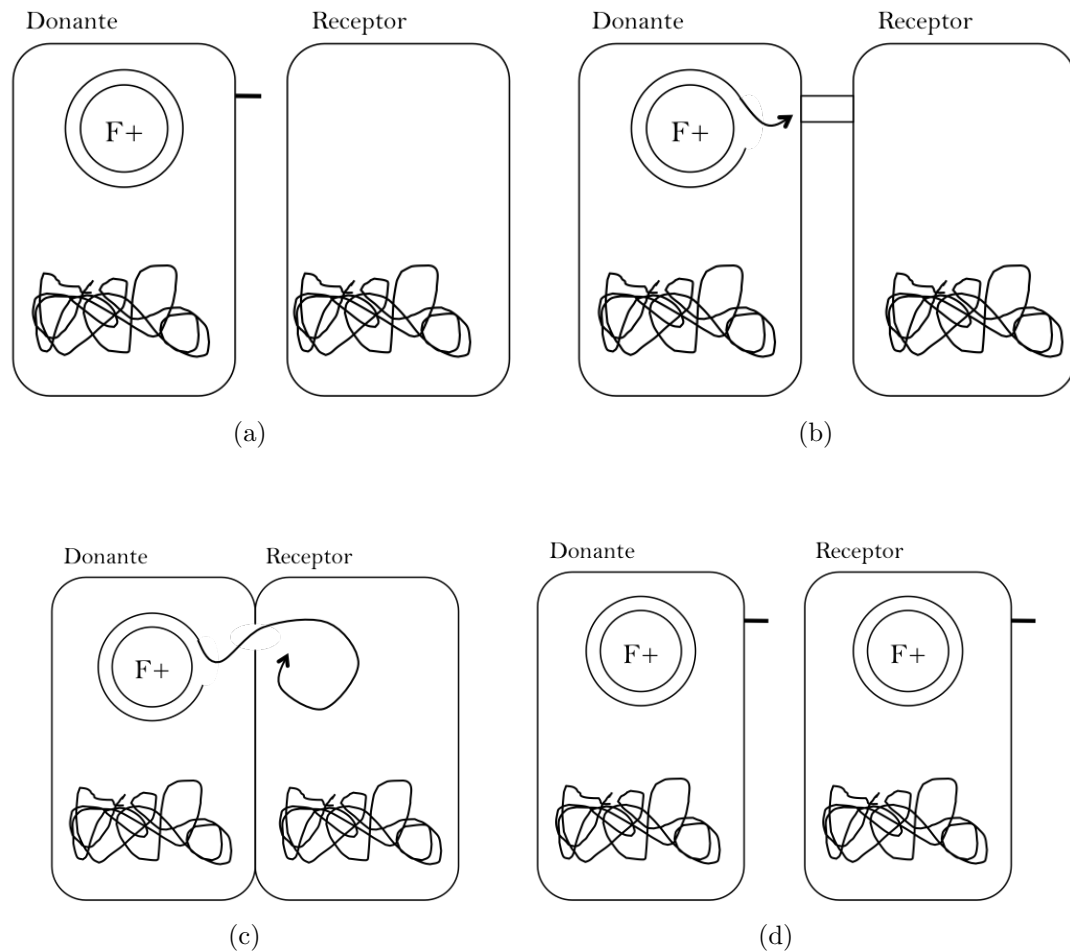


Figura 7.3: Proceso de conjugación bacteriana. a) Aproximación por parte de la bacteria donante a la receptora. b) Contacto por medio del pelo sexual. c) Formación del puente conjugativo y comienzo de la replicación. d) Resultado de la transferencia: dos bacterias donantes.

- En esta segunda etapa representada por la figura 7.3b, la bacteria donante contacta por medio de uno de sus pelos sexuales con la membrana de la bacteria receptora. A continuación, se retrae éste hasta que las células establecen contacto. Durante esta fase, el plásmido característico de las bacterias fértil, el F+, comienza el proceso de replicación.
- Una vez las bacterias han entrado en contacto directo membrana-membrana (figura 7.3c), se forma un puente conjugativo que conecta los citoplasmas de

ambas bacterias para comenzar el intercambio de información genética.

- En la figura 7.3d se observa como al terminar la transferencia las células se vuelven a separar. Ambas bacterias, tanto la donante como la receptora tienen una copia completa del plásmido F+ y quedan listas para protagonizar futuras conjugaciones como donantes.

Una importante característica que es importante recalcar hace referencia al flujo de datos que se produce en la transferencia horizontal. La dirección de la conjugación siempre es la misma y se realiza desde una bacteria fértil a otra infértil. Además, una bacteria fértil nunca más puede ser receptora, lo que marca la dirección del flujo de datos.

7.3. Comunicaciones *Chip-to-Cell*

La manipulación de la comunicación y traspaso de información entre dispositivos electrónicos y organismos vivos, o comunicación chip-to-cell [Sim05], supone un reto quizá algo mayor que los otros mecanismos de comunicación vistos en el capítulo.

El estudio de la comunicación celular desde el punto de vista de la ingeniería desemboca en el manejo de ambas fuentes de información -chip y célula- de manera simultánea, como camino a seguir para crear biodispositivos híbridos aprovechando así las ventajas que ambos materiales puedan proveer³.

Los desarrollos y algoritmos diseñados en esta tercera parte, basan su funcionamiento en la modificación y manipulación de los dos primeros mecanismos de comunicación, el quorum sensing y la transferencia genética horizontal.

³Resulta interesante mencionar otra vertiente de la investigación con bacterias que involucra también a la energía eléctrica: la obtención de corriente eléctrica partiendo del procesamiento metabólico de las bacterias [Sch03] [Nev09] [YiH09]. Se trata aquí de conseguir que pequeños aparatos eléctricos, como las calculadoras, puedan funcionar correctamente alimentándose únicamente de la energía producida por una comunidad de bacterias.

Se propone la comunicación *chip-to-cell* como posible línea de investigación futura para mejorar las capacidades de los dispositivos puramente biológicos.

Capítulo 8

Arquitecturas de comunicación

in-vivo

Índice

8.1. Arquitecturas basadas en señalización <i>QS</i>	70
8.2. Arquitecturas basadas en la conjugación	80

Las sociedades bacterianas presentan una gran variedad de funciones comunicativas que pueden ser manipuladas para servir fines específicos.

El objetivo de la Biología Sintética, en cuanto a diseño de poblaciones se refiere, consiste en alterar las capacidades de los individuos particulares para conseguir bacterias con aplicaciones específicas. De esta manera, las bacterias pueden formar, junto con las funcionalidades de otras células, procesos síncronos y complejos.

Los consorcios de bacterias permiten el estudio de un aspecto fundamental en las Ciencias de la Computación como es la división de tareas. Las poblaciones mixtas así diseñadas pueden llevar a cabo funciones que resultarían muy difíciles, o incluso imposibles, para cepas o especies aisladas. De esta manera, la coordinación funcional entre dos o más poblaciones unidas en un solo organismo puede afrontar problemas hasta ahora insuperables en algunas circunstancias [Bre08].

Como ejemplo de esta colaboración por división de tareas en el que se observa

el beneficio de separar los procesos en cepas, serviría el descrito en [Eit08]. Este desarrollo se centra en la fermentación de azúcares, para lo cual se diseñan dos cepas de *Escherichia coli* de modo que cada una metabolice un azúcar distinto. Como resultado, se concluye que la población mixta formada por las dos cepas fermentan los azúcares de manera mucho más eficiente a la que lo haría una sola cepa con capacidad de metabolizar ambas sustancias.

Otra importante ventaja de la utilización de comunidades heterogéneas con más de dos especies o cepas genéticamente modificadas, es la capacidad de éstas de llevar a cabo aplicaciones que necesiten cierta secuencialidad. Es decir, la capacidad de *programar* las tareas a realizar divididas en varias fases. Dichas fases podrían ser llevadas a cabo por cepas especializadas que comiencen su actividad en instantes de tiempo específicos a lo largo de la computación.

Las características sociales de las bacterias, ya vistas en capítulos anteriores¹, confieren a una comunidad heterogénea de células de una robustez, estabilidad y eficiencia que no tienen los monocultivos o poblaciones uni-cepa. Por ejemplo, estos comportamientos sociales pueden llevar a una comunidad a ser mucho más resistente a la invasión de otras especies [Bur06], lo que puede marcar la diferencia entre el éxito o el fracaso en el diseño de un biodispositivo celular.

En este capítulo se detallan algunos diseños existentes sobre poblaciones bacterianas que, por la naturaleza de las comunicaciones célula-célula que emplean, sirven de punto de partida para los desarrollos propuestos en los capítulos siguientes.

8.1. Arquitecturas basadas en señalización *QS*

El diseño de las comunicaciones célula-célula mediante la modificación de las señalizaciones de los circuitos de quorum sensing, es un método interesante y versátil de construir biodispositivos celulares sintéticos.

Para realizar un estudio formal de las arquitecturas de comunicaciones diseñadas y facilitar el perfeccionamiento de éstas en la presente Tesis Doctoral se

¹Altruismo, canibalismo, cooperación y competición.

propone una abstracción de las arquitecturas en estructuras similares a las redes de neuronas artificiales (diferenciación en capas funcionales).

Así, los componentes a definir a la hora de diseñar uno de estos biodispositivos serían tres:

- Capa de entrada. En esta capa abstracta se engloban aquellas cepas o especies bacterianas que, siendo modificadas genéticamente para servir los objetivos del problema, se encargan de procesar las entradas al biodispositivo.

Como paso posterior, las bacterias de esta capa deben pasar los datos resultados del procesado de las entradas a la capa siguiente. El parámetro más relevante a especificar es el número de cepas (con distintas funcionalidades) con que debe disponer la capa de entrada.

- Capa de salida. El conjunto abstracto resultado de unir una o más cepas bacterianas con funcionalidad final se denomina capa de salida.

Las funcionalidades de estas cepas son catalogadas como finales porque expresan el resultado o salida que se debe obtener de la correcta ejecución de todas las capas previas. Por ejemplo, si el biodispositivo es diseñado para producir luz fluorescente en respuesta a un determinado estímulo, las bacterias que expresan la información lumínica se situarían en esta abstracta capa de salida. Es también de interés el número de cepas que conforman esta capa.

- Capas intermedias. Las cepas definidas en el ámbito de las capas intermedias deben ser aquellas que, ni reciban las entradas del biodispositivo, ni expresen las salidas finales de éste. Por tanto, todas las bacterias con funcionalidades requeridas para la correcta cooperación de las cepas de una comunidad heterogénea y que no efectúen los procesos anteriores, se catalogarán como pertenecientes a una capa intermedia.

Al igual que las capas anteriores, el objetivo a alcanzar referido al número de cepas en la capa intermedia es la minimización de éste sin influir en el correcto funcionamiento del biodispositivo.

- Conexiones. Las conexiones entre las capas hacen referencia al flujo de datos en la población. Debido al carácter difuso de los procesos moleculares y a la gran variedad de conexiones que se pueden establecer, es importante una buena definición previa de éstas.

La correcta especificación de los tres componentes anteriores ayudan a clasificar el diseño de la población bacteriana por su arquitectura de comunicaciones. A su vez, esta formalización facilita la labor de expansión del modelo mediante, por ejemplo, el aumento de capas, la variación del número de cepas en las capas o la cantidad y naturaleza de las conexiones.

Definida de esta manera, una comunidad heterogénea Γ equivaldría a:

$$\Gamma = (L_1, \dots, L_n, U_1, \dots, U_m) \quad (8.1)$$

donde los elementos L_i y U_i representan una capa y una conexión respectivamente.

Una capa es definida de la forma:

$$L_i = (C_1, \dots, C_p) \quad (8.2)$$

siendo p el número mínimo de cepas necesario para la formación de la capa.

Una conexión es definida de la forma:

$$U_i = (C_a \rightarrow C_b) \quad (8.3)$$

en la que el flujo de datos va de la cepa C_a a la cepa C_b . Es importante notar que las conexiones no tienen por qué unir cepas de diferentes capas. Cualquier unión entre cepas es posible en la arquitectura.

Una cepa bacteriana se define de la forma:

$$C_i = (B_1, \dots, B_q) \quad (8.4)$$

donde el parámetro q especifica el número de bacterias B_i que forman una cepa determinada. Este último parámetro es el menos importante en la formalización de la arquitectura y permite cierta flexibilidad.

El diseño de bajo nivel del sistema se realiza en cada bacteria B_i y corresponde con una configuración inicial de ésta B_i^0 . Una computación es definida, por tanto, como el paso de la configuración inicial de una bacteria a la configuración siguiente B_i^1 . Este cambio es detonado por $B_i^0 \vdash B_i^1$.

Es también importante en este proceso de formalización, la diferenciación en estados del biodispositivo dependiendo de la filosofía funcional de éste. Así, se puede distinguir entre:

- Biodispositivo armado. Este estado se alcanza una vez todas las cepas o especies que conforman la población total han sido satisfactoriamente modificadas transgénicamente por separado y se juntan en una misma población.
- Biodispositivo activado. En este estado, la población correspondiente al biodispositivo armado recibe la entrada que estimula la reacción de la capa de entrada.

La distinción entre estos dos estados es importante dependiendo del dominio de aplicación del biodispositivo. Así, habrá ocasiones en las que interese que ambos estados se alcancen al mismo tiempo y situaciones que requieran cierta autonomía del dispositivo (aplicaciones médicas) en los que una diferenciación clara es necesaria.

En la figura 8.1 se ilustran los dos modelos de arquitecturas de comunicaciones, relativamente simples, que siguen los ejemplos presentados a continuación.

El modelo 8.1a se corresponde con una arquitectura en la que el flujo de datos avanza en sentido único y el modelo 8.1b representa una comunidad en la que existen conexiones entre las bacterias de la misma capa.

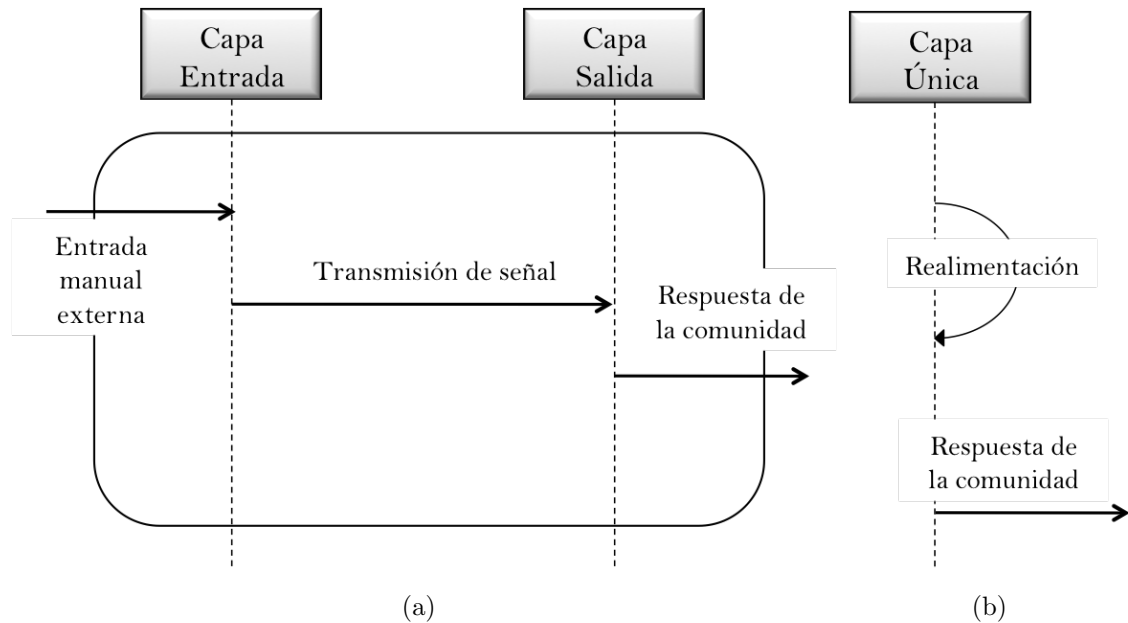


Figura 8.1: Arquitecturas básicas de comunicaciones QS. a) Arquitectura de sentido único y salida simple con dos capas. b) Arquitectura retroalimentada con una capa y salida simple.

8.1.1. Flujo de datos en sentido único

Las comunidades heterogéneas diseñadas para la formación de patrones gráficos [Bas05] [Soh09] [Alb09], constituyen un buen ejemplo de la aplicación de arquitecturas de sentido único (figura 8.1a). De los tres ejemplos propuestos aquí se detalla el primero al ser más sencillo manteniendo todos los componentes necesarios para definir la arquitectura: capas y conexiones.

En este trabajo ([Bas05]) se muestra un sistema multicelular en el que un grupo de bacterias genéticamente modificadas para representar el rol de receptoras, dibujan patrones gráficos lumínicos basados en la concentración de moléculas AHL que produce otro grupo de bacterias denominados emisores.

Dos capas son utilizadas en el biodispositivo:

- Capa de entrada. Esta capa la forma una única cepa bacteriana denominada emisora. Está encargada de recibir una señal de estrés externa a la comunidad

con el fin de comenzar la producción de moléculas AHL.

- Capa de salida. Al igual que la capa anterior, una única cepa bacteriana forma esta capa. Es la denominada cepa receptora y su misión es recibir la señalización oportuna de la capa de entrada y producir, en consecuencia a la naturaleza de la entrada, una salida lumínica que sirve como salida del sistema.

La particularidad de este diseño se encuentra en la disposición física de las cepas que forman el biodispositivo. En lugar de estar mezcladas en la solución nutritiva que sirva como sustrato, la capa de entrada se sitúa en el centro de la población y la capa de salida alrededor de ésta de forma concéntrica.

Es importante el carácter fijo de la capa de entrada, es decir, la situación física de ésta debe permanecer inalterada para producir el efecto deseado.

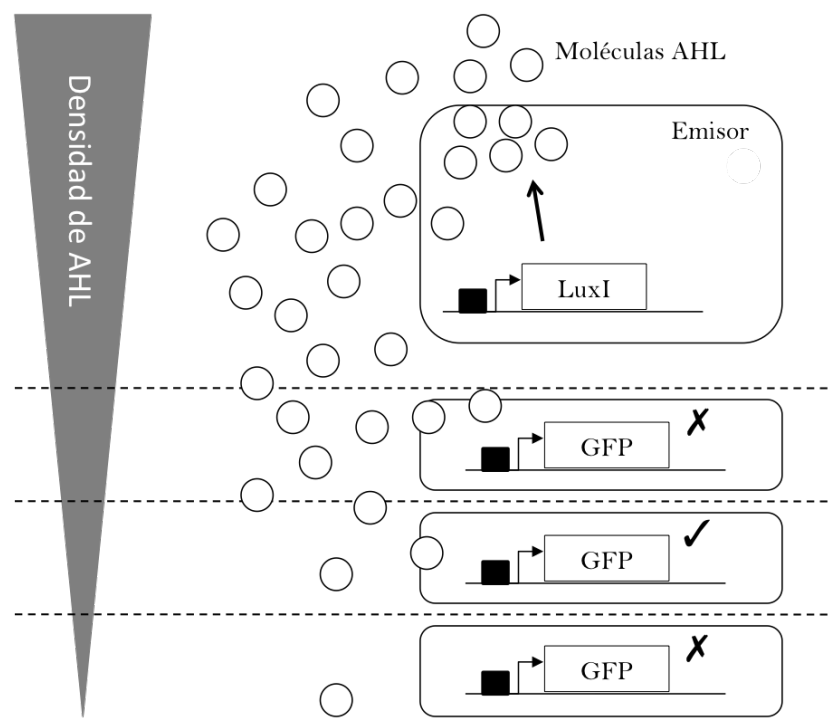


Figura 8.2: Diseño de la comunidad con función de dibujo de patrones. LuxI: gen estructural encargado de iniciar la secuencia de acciones que finaliza en la producción de moléculas de señalización AHL; GFP: Proteína de fluorescencia verde; El gradiente indica de mayor a menor (de arriba a abajo) el nivel de concentración molecular.

La figura 8.2 muestra esquemáticamente el mecanismo que el biodispositivo bacteriano usa con el fin de dibujar patrones lumínicos.

El nivel de producción constante de moléculas AHL por parte de la capa de entrada provoca, como consecuencia, un gradiente en la concentración de dichas moléculas en la comunidad. De esta manera el nivel de concentración en el centro de la comunidad, donde se sitúa la capa de entrada, es muy superior al que se puede medir en el extremo de la población.

Los dibujos originados por la capa de salida están relacionados con este gradiente del nivel de concentración molecular de AHL. Así, la programación genética de las bacterias de esta capa se comporta de la siguiente manera respecto al nivel de densidad de moléculas de señalización formando bandas de detección:

- Nivel de densidad molecular alto. Las bacterias de la capa de salida *sienten* un nivel de señalización alto que supera ampliamente el umbral correspondiente a su membrana celular. Como consecuencia de este fenómeno, la actividad en el interior de las bacterias se subdivide en dos tareas concurrentes:
 1. Se desbloquea la represión (programada por omisión) de GFP². Esta acción necesita que el umbral se supere levemente.
 2. Por otro lado, se bloquea nuevamente la expresión de GFP mediante otro mecanismo que necesita que el umbral se supere ampliamente para ser ejecutado.

Por consiguiente, la expresión lumínica es reprimida en las bacterias de la capa de salida más cercanas al centro de la población.

- Nivel de densidad molecular medio. Estas bacterias también ven su umbral superado por el nivel de moléculas AHL en el ambiente. A diferencia del nivel previo, el umbral es sólo superado levemente por lo que sólo se activa un proceso de los dos posibles dentro de la bacteria: el desbloqueo de la expresión del gen de fluorescencia. Como consecuencia, estas bacterias lucen.

²Green fluorescent protein o proteína de fluorescencia verde.

- Nivel de densidad molecular bajo. Las bacterias de la capa de salida que se sitúan en un lugar más lejano al centro de la población *sienten* un nivel de concentración muy bajo en su entorno, el cual no es suficiente para disparar la acción de la bacteria. La célula queda, por tanto, en el estado inicial y la luz reprimida.

Siguiendo esta metodología pueden dibujarse circunferencias en un color mientras que su centro es de otro, distinguiéndose así las dos capas del biodispositivo.

La figura 8.3 muestra los resultados de combinar la acción de varias capas de entrada situadas en puntos distintos de la población. Se crea así dibujos representativos de la unión de las circunferencias concéntricas a cada capa de entrada.

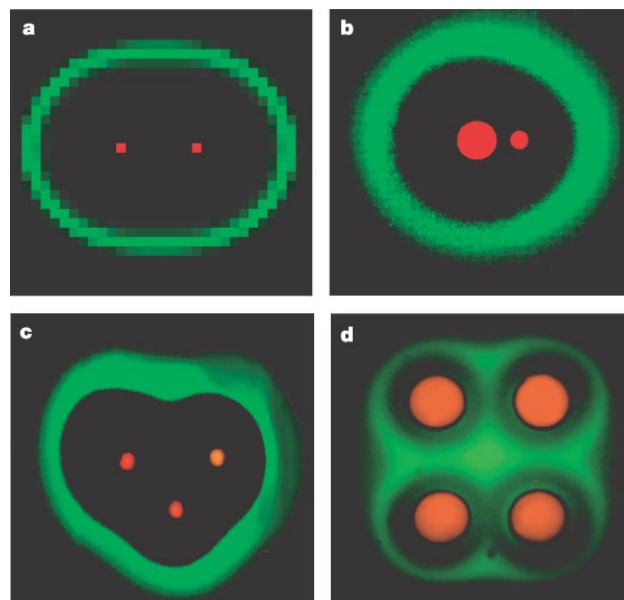


Figura 8.3: Formación de varios patrones. a) Simulación computacional del comportamiento del biodispositivo. b)-d) Resultados experimentales. b) Formación de una elipse. c) Formación de un corazón. d) Formación en flor. Extraída de la web http://www.nature.com/nature/journal/v434/n7037/fig_tab/nature034s61_ft.html.

Otra ejecución de este diseño en la que se utilizan señales lumínicas diferentes en cada banda de detección se observa en la figura 8.4, donde las bacterias de la capa de salida son programadas para expresar un color diferente dependiendo del nivel de concentración de moléculas AHL en su entorno.

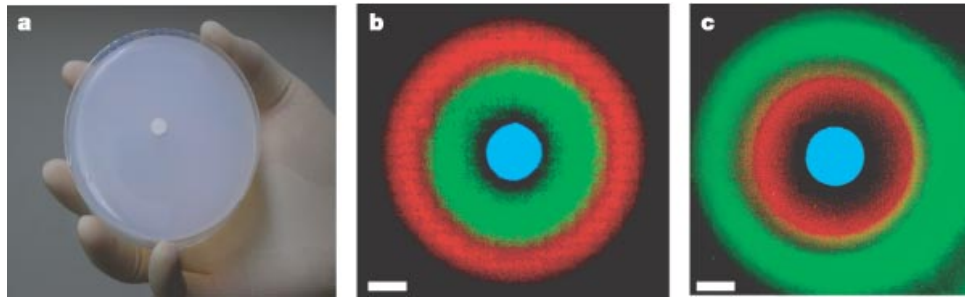


Figura 8.4: Bandas de detección con expresiones lumínicas distintas. a) Placa de Petri con la cepa emisora situada en el centro. b)c) distintos dibujos resultado de la expresión de diferentes proteínas de fluorescencia. La cepa emisora expresa siempre CFP. Extraída de la web http://www.nature.com/nature/journal/v434/n7037/fig_tab/nature034s61_ft.html.

Como consecuencia a esta última ejecución, se dibujan en la población heterogénea círculos concéntricos de diferentes colores gracias a la expresión de proteínas de fluorescencia distintas.

Las características de esta arquitectura son ampliamente mejoradas en los desarrollos propuestos en capítulos siguientes, explorando en éstos tanto el aumento de cepas con funcionalidades distintas en la misma capa como el número y sentido de las conexiones.

8.1.2. Autoseñalización

Como ejemplo de arquitectura de autoseñalización se estudiará una comunidad bacteriana heterogénea diseñada, según la formalización del diagrama 8.1b, para conseguir una funcionalidad lógica AND [Bre07].

En esta comunidad, dos cepas bacterianas de *Escherichia coli* son modificadas genéticamente para intercambiar moléculas AHL en una conversación bidireccional cuyo fin es la notificación de la densidad de población de ambas. Debido a la programación interna de las bacterias, este consorcio molecular expresará los genes objetivo que marcan el fin exitoso de la computación sí y solo sí ambas cepas están presentes por encima de determinada población, alcanzando así una funcionalidad AND.

En relación con la formalización propuesta, es ésta una arquitectura con una

sola capa formada por dos cepas bacterianas. Ya que la filosofía funcional del biodispositivo funde los dos estados posibles³ en uno solo, esta comunidad no tendrá entrada externa y se autoregulará desde el momento de formación de ésta.

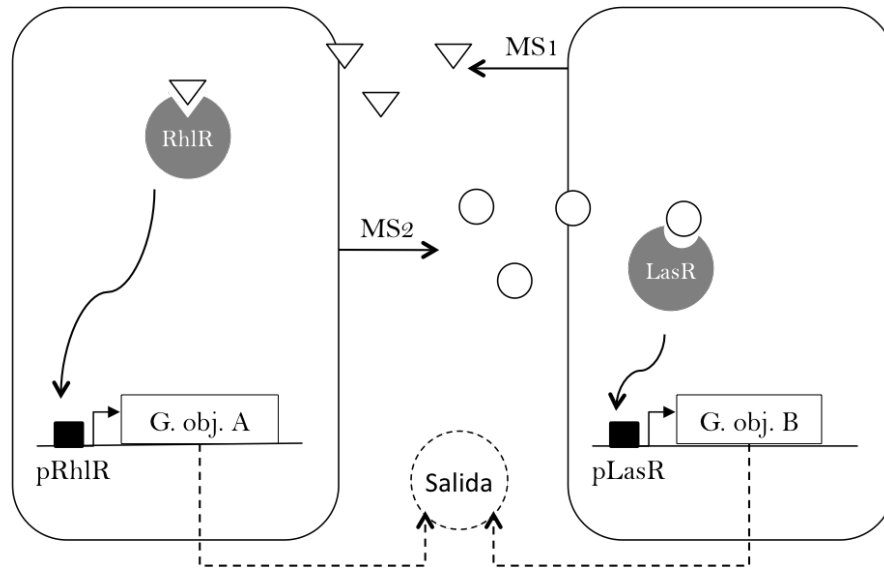


Figura 8.5: Consorcio celular con funcionalidad lógica AND. Las dos poblaciones de *E. coli* se comunican usando el mecanismo de quorum sensing de la bacteria *P. aeruginosa*. MS₁: molécula de señalización *C4HSL* producida por una de las cepas y recibida por RhIR; MS₂: molécula de señalización *3OC12HSL* producida por la otra cepa y recibida por LasR; *G. obj.*: genes objetivo.

La figura 8.5 muestra el esquema de comunicación que usan las dos cepas del biodispositivo, el cual está basado en el mecanismo de quorum sensing de la bacteria *Pseudomonas aeruginosa* visto en el capítulo anterior. Mediante los pares LasI/LasR y RhII/RhIR de dicho circuito, el diálogo entre las dos cepas se lleva a cabo para que cada una pueda informar en todo momento de su nivel de concentración a la otra.

Por lo tanto, la densidad de la concentración de ambas cepas debe ser suficiente para que los dos pares de proteínas que sirven como mecanismo de comunicación puedan inducir los genes objetivo de las bacterias. En este caso, si los genes objetivos codifican información lumínica diferente, resultará fácilmente comprobable si

³Recordar los dos estados en los que el biodispositivo puede encontrarse: armado y activado.

el funcionamiento AND actúa como debería. Si es así, la comunidad expresará un color que es mezcla de ambos.

Aún aumentando el número de cepas por capa, la presente arquitectura resulta relativamente simple ya que el hecho de que la salida del sistema sea única, marca límites estrictos al biodispositivo. El número de salidas representa un factor determinante para la autonomía funcional de la comunidad y, como tal, constituye una de las características fundamentales del último desarrollo de esta Tesis Doctoral: el oscilador multicelular.

8.2. Arquitecturas basadas en la conjugación

El otro paradigma de comunicación expuesto a estudio es la transferencia horizontal de información genética entre bacterias (conjugación), cuyo mecanismo básico se ha detallado en capítulos anteriores.

Así como se encuentran con relativa facilidad en la bibliografía actual ejemplos de algoritmos matemáticos que se inspiran en esta técnica [Mon09] con la finalidad de optimizar los costes computacionales en la resolución de problemas complejos, es más difícil encontrar ejemplos que utilicen la conjugación *in-vivo* y su modificación como medio de alcanzar objetivos dirigidos.

De entre estos últimos, se selecciona el experimento de Wakabayashi [Wak05] para ser explicado en esta sección. Los motivos para esta selección son los siguientes:

- Explora una característica de la conjugación bacteriana que tiene que ver más con el flujo de datos en la comunidad celular que con la codificación de información en los plásmidos.

La computación con plásmidos, ya sea *in-vitro*, *in-vivo* o *in-info*, se ha desarrollado tradicionalmente manipulando las cadenas de ADN de éstos para encriptar los datos o metadatos del problema a resolver. Este es el procedimiento también del modelo propuesto en el capítulo siguiente⁴ en el que se deja al libre albedrío la circulación de plásmidos entre las bacterias de la población.

⁴Se propone aquí una metodología que, basándose en procesos de conjugación, permite la reso-

Sin embargo, el ejemplo detallado a continuación se centra en desarrollar un modelo para sincronizar la circulación de plásmidos sin que ésta se realice al azar. De esta manera se consigue, de una forma básica, dirigir el flujo de datos en la comunidad.

- Aunque el modelo de conjugación visto en el capítulo anterior es el más genérico, en este ejemplo se manipulan los mecanismos de conjugación de la especie bacteriana *Enterococcus faecalis*.

E. faecalis realiza procesos de transferencia genética en el sentido *célula donante* \rightarrow *célula receptora* pero presenta una inicialización del proceso radicalmente distinto. En este caso, no es el donante el que se acerca inicialmente al receptor para establecer el contacto, sino que es el receptor el que inicia esta relación mediante la producción de feromonas que atraen al donante. Todas las bacterias, tanto donantes como receptoras, segregan feromonas. Sin embargo, las bacterias donantes segregan, gracias a la acción del plásmido conjugativo, inhibidores de éstas.

Por las dos razones anteriores este experimento supone un interesante punto de vista, en cuanto a la conjugación bacteriana se refiere, que ayuda a completar la especificación de este mecanismo de comunicación.

El sistema de conjugación de *E. faecalis* es aplicado en el ejemplo citado a la construcción de puertas lógicas orientadas a la computación con bacterias. Las feromonas *C* y los inhibidores *I* se utilizan como señales de control que permiten, o no, el flujo de datos (traspaso de plásmidos) de una célula a otra.

La figura 8.6 muestra una de estas puertas lógicas básicas implementada con células *in-vivo*. Dependiendo del estado de las señales de entrada, el plásmido *P* es liberado o no de la célula emisora. En presencia de *C* y ausencia de *I* el plásmido inicia el proceso de replicación hacia la célula receptora.

lución del problema SAT. Como se verá más adelante, se codifican en los plásmidos las variables de la fórmula lógica a resolver.

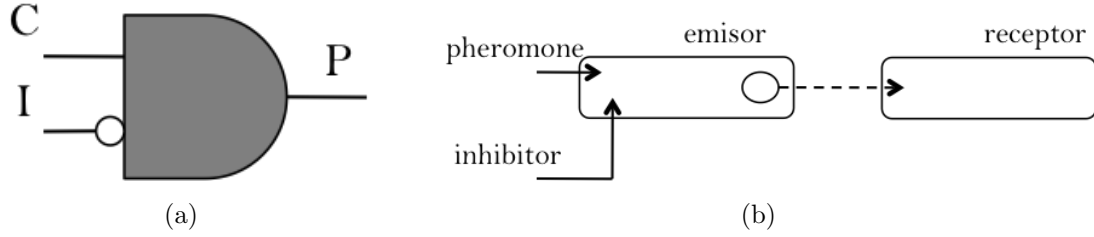


Figura 8.6: Esquema de una puerta de información básica siguiendo el mecanismo de la conjugación bacteriana. a) Diagrama lógico. C: feromonas; I: señales inhibitoras; P: plásmido. b) Esquema funcional.

El comportamiento aparentemente lógico (*ON*, *OFF*) de estas células se combina, en el mismo experimento [Wak05], con el fin de construir el inversor lógico de la figura 8.7.

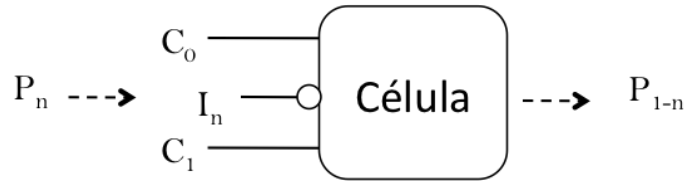


Figura 8.7: Esquema del inversor lógico controlado por el mecanismo de transferencia genética horizontal de la bacteria *E. faecalis*. C_i : feromonas; I_n : señales inhibitoras; P_i : plásmido conjugativo.

Los receptores tienen dos tipos de donantes posibles: donantes con plásmidos P_0 y donantes con plásmidos P_1 . El control sobre qué donante realiza el proceso de conjugación con la célula receptora lo lleva a cabo las sustancias inhibitoras, de tal forma que al recibir la molécula inhibidora I_n los donantes con plásmidos P_{1-n} son activados. La bacteria receptora produce entonces nuevas señales inhibitoras dependiendo de la información genética del plásmido transferido. Estas nuevas señales de control servirán como entrada de la siguiente operación.

El desarrollo propuesto en el capítulo siguiente utiliza la conjugación como fin para realizar computaciones en una población bacteriana, diseñando para ello, la información que contienen los plásmidos. Sin embargo, el experimento aquí planteado trata de modelar el flujo que los plásmidos siguen entre las bacterias. Por ello, este

modelo puede influir positivamente en futuros desarrollos de biodispositivos en los que la conjugación juegue un papel fundamental.

La creación en un futuro no muy lejano de proteínas sintéticas y células mínimas⁵, todas ellas dirigidas a la realización de una tarea específica, contribuirán sin duda a perfeccionar los biodispositivos y optimizar sus características.

⁵Se encuentra una breve introducción al concepto de célula mínima en el capítulo siguiente

Capítulo 9

Modelo de resolución del problema SAT

Índice

9.1. Células mínimas	86
9.2. Diseño del algoritmo	89
9.3. Codificación genética	96
9.4. Computando una instancia del SAT	99

En este capítulo se presenta un modelo de computación paralela basado en la comunicación entre las bacterias de una población multi-cepa.

De las dos formas de comunicación estudiadas en este modelo se hace uso exclusivamente de la comunicación por conjugación o transferencia genética para desarrollar un algoritmo *in-vivo* y autónomo que resuelva una instancia del problema SAT (satisfiability). El objetivo de este algoritmo es la mutación dirigida de un conjunto de plásmidos inicial a un conjunto final que representará la solución al problema.

Una instancia del problema SAT es una fórmula booleana que tiene tres componentes [Jun96]:

- Un conjunto de n variables: x_1, x_2, \dots, x_n .
- Un conjunto de literales.

Un literal es una variable ($Q = x$) o la negación de una variable ($Q = \bar{x}$).

- Un conjunto de m cláusulas distintas: C_1, C_2, \dots, C_m .

Cada cláusula consiste la unión literales combinados por conectores lógicos *or* (\vee).

El objetivo del problema SAT is determinar si existe o no una asignación concreta de valores (*True* o *False*) a los literales de manera que se valide la fórmula:

$$C_1 \wedge C_2 \wedge \dots \wedge C_m \quad (9.1)$$

donde \wedge es el conector lógico *and*.

La conjugación confiere a las bacterias una característica única entre los seres vivos: la transferencia genética horizontal.

Al final de los años 80 y durante la intensificación en el trabajo de novedosas técnicas de secuenciación bacterianas y familias de microorganismos unicelulares, varios científicos [Gog89] se empezaron a fijar en cómo los genomas de las bacterias estaban mucho más mezclados y eran más uniformes de lo que a priori habían pensado. En la década siguiente los investigadores en biología molecular continuaron observando la misma tendencia por parte de las poblaciones de bacterias a tener una información genética muy similar en la misma generación [Gog02]. Sin duda, esto representaba una contradicción con la visión metafórica del *árbol de la vida*.

La transferencia genética horizontal significa una nueva forma de organización en la que poder etiquetar la actividad generacional de las bacterias. Las investigaciones previamente citadas muestran que la información genética no sólo se traspasa de padres a hijos de manera vertical entre una generación y su generación descendiente, sino que también es posible encontrar en la naturaleza seres vivos como las bacterias que intercambian información genética entre los individuos de una misma generación.

Esta transferencia horizontal se realiza incluso entre individuos de distinta especie y, a veces, entre distintos taxones en que actualmente se considera subdividida la diversidad de los seres vivos: arqueas (*Archaea*), bacterias (*Bacteria*) y eucariontes (*Eukarya*).

Ya que la evolución fue inicialmente estudiada en animales y plantas, la tendencia habitual ha sido considerar la transferencia genética como un fenómeno vertical. Sin embargo en el mundo bacteriano este paradigma no explica de la mejor manera la actividad genética que se lleva a cabo, en la que la información genética relevante a la supervivencia de la especie se transmite entre individuos sin necesidad de esperar a la generación siguiente.

El diseño de la comunidad heterogénea de bacterias presentado en este capítulo se fundamenta en el uso de este modelo de transferencia para realizar computaciones dirigidas a objetivos concretos. Esta comunidad está formada por varias cepas bacterianas modificadas transgénicamente.

La idea principal del algoritmo diseñado está basada en la separación y diferenciación de la información genética por la funcionalidad de la misma.

Se distingue entre los datos del problema a resolver, los cuales estarán codificados en plásmidos o vectores, y las instrucciones computacionales, codificadas en regiones específicas del cromosoma bacteriano.

Ambas informaciones se emplean en procesos distintos como se verá en secciones sucesivas y es su combinación la que otorga efectividad al algoritmo computacional.

9.1. Células mínimas

En paralelo al desarrollo y diseño de bacterias con funcionalidades diversas, se está llevando a cabo un importante trabajo de búsqueda del contenido mínimo del genoma celular [Moy09]. Esta búsqueda trata de encontrar la información genética básica de la célula que la permita vivir, alimentarse y duplicarse con la finalidad de poder sintetizar esa información y crear células *ex-novo*.

Aunque muy difícil, este reto supondría un paso muy importante en la computación con bacterias ya que la capacidad de reutilización de éstas y el control sobre sus ciclos metabólicos conferirían a los diseños basados en biodispositivos celulares de una gran eficacia y robustez.

Los desarrollos de esta Tesis Doctoral están especificados para su implantación en microorganismos sintéticos diseñados *a la carta* aprovechando las características antes mencionadas para conseguir una máxima viabilidad.

Las bacterias son, desde un punto de vista computacional, seres bastante complejos con gran cantidad de información codificada en su cromosoma y una alta variedad de sofisticadas rutas metabólicas. Si además de las funciones normales que desempeñan se añeden las funcionalidades propias de los diseños detallados en este capítulo y siguientes, el control absoluto sobre el comportamiento esperado resultaría más complejo debido al carácter difuso de los procesos moleculares. Por esto, el uso de bacterias mínimas sintetizadas *ex-novo* que contaran con un conjunto de funcionalidades muy reducido, sería muy ventajoso para reducir al máximo la interacción de procesos celulares indeseados con las computaciones para las cuales se diseñan estos seres vivos.

Con la finalidad de establecer una escala numérica para que el lector se familiarice con el tema en cuestión basta con echar un vistazo a uno de los últimos genomas bacterianos que se ha secuenciado exitosamente en su totalidad: el genoma de la especie *Actinosynnema mirum*¹ [Lan09] que cuenta con 8,248,144 pares de bases. Teniendo en cuenta que un gen de *gfp*² ocupa aproximadamente 700 pares de bases, se puede tener una idea de la ínfima información genética extra que los desarrollos aquí propuestos suponen con respecto al genoma total de la bacteria.

Una visión clásica de las ventajas que se desean obtener mediante el diseño de células mínimas [Mor92] dividiría las funcionalidades a obtener en dos facetas:

¹La *Actinosynnema mirum* es una bacteria de la familia de las actinobacterias o actinomycetes. Esta familia de células Gram positivas se caracterizan por tener un alto ratio de G + C.

²GFP, green fluorescent protein o proteína de fluorescencia verde.

- Minimización de la red metabólica esencial para interactuar con el entorno y servir sus ciclos celulares internos.

Computacionalmente, esta faceta equivaldría a minimizar la carga de proceso en cada procesador de una red distribuida.

- Minimización de las necesidades de abastecimiento del entorno (por ejemplo CO₂, N₂ y minerales)

Computacionalmente equivaldría a optimizar el uso de recursos en cada procesador.

Ambas ventajas resultan de importancia para poder potenciar las características de las bacterias diseñadas con fines computacionales.

Sin embargo, la posibilidad de síntesis de células mínimas vivas todavía queda lejana debido al actual desconocimiento de gran parte del ciclo celular. Los organismos biológicos, incluso el más sencillo de los organismos unicelulares, son altamente complejos.

Analizando únicamente la información genética representada por los genes que codifican proteínas, alrededor de una tercera parte de este conjunto, independientemente del genoma que se escoja, tiene una funcionalidad desconocida. En el caso de la especie bacteriana *Actinosynnema mirum* el genoma cuenta con 7100 genes de proteínas lo que significa que aproximadamente 2367 proteínas de esta bacteria desempeñan un trabajo que se desconoce.

Esta enorme falta de conocimiento sobre los agentes moleculares que conforman una célula (sólo se ha hablado de los genes de codificación de proteínas), hace que resulte muy difícil la tarea de predicción del comportamiento de cualquier organismo así como estructurar, catalogar y caracterizar sus componentes en función de su uso.

La búsqueda del conjunto de genes mínimos se realiza, por tanto, mediante un algoritmo de prueba y error en el que se observa el comportamiento de las células tras varios procesos consistentes en retirar genes del cromosoma.

Una vez establecido el conjunto genético mínimo [Moy09] que presente funcionalidades vitales completas para un sistema específico, el cromosoma de la futura bacteria artificial que contenga dicho conjunto mínimo puede obtenerse mediante síntesis de ADN, e introducido posteriormente en un compartimento celular para fabricar células mínimas.

De estar bien configurados estos organismos vivos artificiales serían contenedores ideales para realizar pruebas de inserción de nuevos módulos genéticos y diseños con fines computacionales para moldear nuevos genomas que caractericen microorganismos con diferentes funcionalidades.

La forma de diseñar y modelar estos genomas sintéticos es también muy extensa, al menos desde el punto de vista de las posibilidades de codificación de información genética. Por ejemplo, la incorporación de aminoácidos no naturales en las proteínas celulares es un campo apenas sin explorar en la ingeniería molecular. Añadiendo varios aminoácidos no naturales se permite la fabricación de novedosas proteínas con nuevas y dirigidas funciones [Dru07].

El diseño detallado a continuación enfatiza la utilidad de los plásmidos y su transferencia por conjugación. El uso de este protocolo de comunicación encaja perfectamente con la filosofía de utilización de células mínimas ya que aquellos genes necesarios para llevar a cabo los procesos de conjugación están incluidos en el genoma mínimo, tal y como se concluye de recientes resultados [Per09]. La capacidad intacta de realizar procesos de transferencia horizontal de genes durante la evolución de células mínimas indica la importancia y utilidad que el uso de plásmidos puede tener en desarrollos sintéticos.

9.2. Diseño del algoritmo

El objetivo de este algoritmo es proveer a una comunidad bacteriana multi-cepa con la capacidad suficiente para modificar y evolucionar información genética con el fin de adecuar ésta a un objetivo buscado. La información genética manejada está dividida en dos tipos: cromosomas y plásmidos.

Recordando los modelos de computación *in-vitro* en los que se usaban plásmidos, se observan varias similitudes con el algoritmo diseñado en este capítulo. Estas semejanzas abarcan el uso de plásmidos como estructuras de datos útiles para ser manipuladas.

En dicho proceso de manipulación es donde se diferencian los dos métodos: el método *in-vitro* utiliza manipulación manual para hacer evolucionar los datos y el algoritmo propuesto que se basa en la actividad celular como fuente de manipulación autónoma en la computación *in-vivo*. Esta traducción de *in-vitro* a *in-vivo* es el objetivo fundamental que persiguen no sólo los desarrollos de esta Tesis Doctoral sino los últimos avances en Biología Sintética, tratando de encontrar así aplicaciones en campos de investigación tan diversos como la ecología o la medicina.

Después del diseño de los plásmidos en los que debe ir codificada toda la información manejada por el problema (datos iniciales), los algoritmos diseñados en los experimentos tradicionales protocolizaban una serie de pasos de clonación de la población en diferentes tubos de ensayo, en los que se llevaban a cabo fases de selección de los cassettes del plásmido presente en el tubo, mediante la introducción en éste de enzimas específicas. Con este procedimiento, el espacio de búsqueda del problema es explorado totalmente representando cada individuo en un plásmido, con lo que sólo queda un último paso de selección para obtener el plásmido buscado.

Con la intención de trasladar el algoritmo de manipulación manual anterior basado en tubos a un nuevo paradigma de computación natural autónomo, se usan cepas bacterianas que emulen la utilización de tubos de ensayo en la fase de exploración del espacio de búsqueda del problema a resolver. Como resultado, la intervención humana se reduce al máximo durante la fase de proceso del problema ya que las cepas interactúan entre sí desempeñando las tareas que se deben realizar manualmente en la computación *in-vitro*.

Las características esenciales del uso genético son dos:

- Transferencia genética de los datos del problema.

La conjugación bacteriana es habilitada³ en la comunidad multi-cepa y los datos son codificados en plásmidos de manera que cualquier bacteria pueda traspasar sus plásmidos a otra. Este proceso simula el envío de datos entre los procesadores de una red.

- Instrucciones cromosómicas.

Siguiendo la analogía en la que consideramos cada bacteria como un procesador independiente, los datos sufren diferentes mutaciones y cambios en cada nodo de la red debido a la actividad de las instrucciones que cada procesador tenga programadas. Estas instrucciones computacionales, que no son más que la información necesaria para llevar a cabo un proceso determinado, son propias de cada procesador de la red. Es decir, son propias de cada bacteria de la comunidad y están almacenadas en el cromosoma celular, que es el lugar donde la información genética local (propia de la célula) se codifica para evitar su intercambio.

En el proceso de traducción de la computación *in-vitro* (con tubos de ensayo) en paradigmas autónomos se observa con facilidad la efectividad de estas instrucciones.

En la figura 9.1.a) se observa la metodología clásica en la que, teniendo muchos plásmidos en un mismo tubo, se introduce manualmente en la solución una enzima capaz de cortar un sector específico de éstos. La misma operación se ilustra en la figura 9.1.b) pero esta vez en un contexto autónomo *in-vivo*: la bacteria superior tiene la capacidad de producir una encima que corta el mismo sector *a* del plásmido. Dicho plásmido, cuando recaiga en la bacteria inferior, sufrirá el corte por un nuevo sitio correspondiendo a la encima que produce esta última bacteria. Así, el plásmido sufre dos reducciones en el mismo cultivo bacteriano y sin intervención humana.

³El conocimiento sobre los procesos moleculares y la alta tecnología permiten hoy en día la inhabilitación de la capacidad de transferencia genética en una población. En los desarrollos de capítulos posteriores, se inhabilitará para evitar comportamientos indeseados.

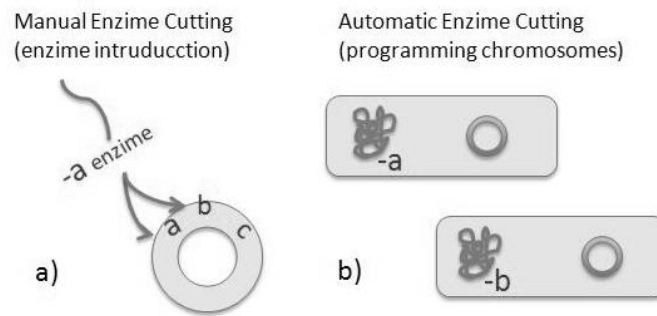


Figura 9.1: Manipulación automática con enzimas. a) Procedimiento manual *in-vitro*; b) Procedimiento automático *in-vivo*.

Este paradigma de computación que aquí se propone sirve de punto de partida para diseñar bacterias que solucionen problemas más complejos. Hay que tener en cuenta que los cromosomas de las células que pertenecen a la misma cepa bacteriana llevan el mismo diseño y, por consiguiente, esas células presentan la misma funcionalidad.

En el ejemplo modelado en este capítulo, se utilizan dos instrucciones por cromosoma:

- ▷ Una instrucción para modificar los plásmidos que circulan por la comunidad mediante ciclos de recombinación dirigida. De esta manera cada bacteria esta encargada de evolucionar el conjunto de datos explorando así el espacio de búsqueda.
- ▷ Una instrucción de evaluación es la encargada de comprobar si el nuevo individuo recombinado cumple o no cierta condición de fin. Si se cumple la condición específica, el conjunto de datos será evaluado satisfactoriamente y representará la solución final mediante la expresión de un *reporter* que permita la fácil identificación de ésta.

Se debe prestar especial atención a las características de los plásmidos usados en este diseño debido al requisito de este algoritmo referido a la separación total de material genético en dos tipos: cromosomas y plásmidos.

Esta independencia necesaria de los dos tipos de información obliga a evitar el uso de los llamados plásmidos integrativos que son aquellos que poseen la capacidad de romper momentáneamente el cromosoma celular e insertarse dentro de éste. Esta funcionalidad que tan útil resulta en otros ámbitos para la posterior replicación de la información genética codificada en el plásmido durante el proceso de fisión celular, es capaz de terminar con la efectividad del algoritmo.

Además de evitar el uso de plásmidos integrativos o episomas es importante atender a su capacidad de conjugación y movilidad para evitar el empleo de:

- Plásmidos conjugativos pero no movilizables (Tra+ Mob-).

Estos plásmidos permiten realizar el contacto entre las bacterias pero no tienen capacidad para transferir su ADN. Como este algoritmo basa su desarrollo en el traspaso de los conjuntos de datos, estos plásmidos no serían de utilidad.

- Plásmidos no conjugativos pero movilizables (Tra- Mob+).

Al contrario que el caso anterior, estos plásmidos si tienen la capacidad de transferir su ADN pero no de crear el medio propicio para ello. Por tanto, resultan igual de inapropiados para el algoritmo.

Los plásmidos que tienen ambas características, conjugativos y movilizables (Tra+ Mob+) se denominan auto-transferibles y son los vectores de uso conveniente en la comunidad heterogénea diseñada. Futuros desarrollos que sigan esta línea de investigación se pueden fundamentar en la combinación de distintos tipos de plásmidos para desarrollos más complejos.

9.2.1. Formalización

El biodispositivo modelado para realizar computaciones por conjugación es una comunidad C definida por el siguiente conjunto:

$$C = \{S_1, \dots, S_i\} \quad (9.2)$$

donde las variables S simbolizan cepas distintas y el parámetro i el número final de éstas que será necesario para resolver el problema tratado.

Una cepa se representa con una estructura de datos similar:

$$S = \{B_1, \dots, B_j\} \quad (9.3)$$

donde las variables B son los objetos de procesamiento mínimos en los que se descompone el biodispositivo, es decir bacterias independientes y el parámetro j el número de éstas en una cepa, con $j \gg i$.

Teniendo en cuenta la jerarquía del diseño y a la vista de los conjuntos anteriores, define la comunidad C de la forma:

$$C = \{B_{11}, B_{12}, B_{13}, \dots, B_{ij}\} \quad (9.4)$$

Según la descripción detallada al principio de la sección para la resolución de un problema SAT, se ha de definir tres características básicas específicas de cada bacteria:

- Un conjunto $[p_n]$ donde se enumeran los plásmidos o conjuntos de datos que representan las posibles soluciones presentes en la bacteria B_{ij} en un instante de computación determinado.
- Una instrucción computacional $[I_i]$ de común implantación en todas las bacterias B_{ij} de la misma cepa S_i , y que especifica el tipo de recombinación que la célula lleva a cabo sobre los plásmidos.

En conjunto, todas las instrucciones $[I]$ deben abarcar todos los cambios posibles en los conjuntos de datos de manera que el espacio de búsqueda del problema sea explorado por completo. El número de estas instrucciones determinará en este ejemplo el número de cepas a usar en la computación.

- Una instrucción computacional de evaluación $[E]$ encargada de evaluar constantemente los conjuntos de datos que se encuentran en la bacteria en un momento determinado. En caso de que el plásmido sea validado por representar el conjunto solución, esta instrucción debe ser capaz de informar de esta confirmación⁴.

⁴En el caso de una confirmación digital, lo más habitual sería usar proteínas de fluorescencia para avisar del hallazgo de solución en una célula.

Se define una bacteria en el estado inicial de la computación como:

$$B_{ij}^0 = \{\emptyset, [I_i], [E = False]\} \quad (9.5)$$

donde el conjunto vacío representa la ausencia de conjuntos de datos en la célula y la instrucción de evaluación no produce respuesta afirmativa.

Esta configuración inicial B_{ij}^0 será igual en todas las células de la comunidad (cada una con la instrucción $[I]$ que corresponda) salvo las células encargadas de comenzar la distribución de plásmidos. Éstas no presentarán un conjunto vacío sino los plásmidos iniciales cuya codificación represente todos los datos del problema.

El uso de una cepa sencilla, sin instrucciones codificadas en el cromosoma ni caracterización especial, cuyo único fin sea llevar los plásmidos necesarios para el comienzo de la computación puede conferir al biodispositivo un interesante carácter de *activación*. Así se podría distinguir entre dos estados:

- Biodispositivo armado. Este estado aparece una vez que todas las cepas que participan activamente en la computación son mezcladas en una misma superficie o medio de cultivo, y en él la comunidad está lista para llevar a cabo el proceso de evolución genética tan pronto como aparezcan plásmidos.
- Biodispositivo activado. Una vez la comunidad en estado armado recibe los plásmidos que codifican los datos del problema, se activa el funcionamiento de ésta y se considera en estado activado.

En futuros desarrollos en los que se aplique este algoritmo en distintos dominios científicos como la ecología, esta visión de la actividad de la comunidad heterogénea dividida en dos estados distintos puede resultar de mucha utilidad. Esto es debido a que esta visión potencia el carácter autónomo al poder dejar la comunidad en estado armado en un medio específico a la espera de ser activada por causas externas.

En cualquier caso, a medida que los procesos de conjugación bacteriana tienen lugar y los conjuntos de datos sufren diversas recombinaciones en el interior de las células, alguna alcanzará el estado final de la computación definido por:

$$B_{ij}^n = \{[p_n], [I_i], [E = True]\} \quad (9.6)$$

donde el conjunto de datos codificado en el plásmido p_n representará el conjunto solución al problema que se pretende resolver.

9.3. Codificación genética

Una especificación más detallada de la codificación con el fin de resolver una instancia del problema SAT se ofrece en esta parte. Debido a las características específicas del problema SAT: 1) La instrucción de evaluación debe ser una función de validación lógica similar a los circuitos genéticos básicos ya vistos en capítulos anteriores; 2) El plásmido inicial debe contener un valor (*True/False*) para cada variable del enunciado; y 3) La instrucción encargada de mutar el plásmido con la finalidad de alterar los valores de las variables está basada en un sistema de intercambio de casetes cuya función se detalla más adelante.

Plásmidos codificados. Para saber qué información es necesario codificar, se han de definir previamente los datos del enunciado. La función lógica f que debe ser validada en la resolución de una instancia del problema SAT está compuesta por una serie de n variables: $f(x_1, x_2, \dots, x_n)$ en la que cada variable x_i puede tomar los dos valores posibles $\{x_i, \neg x_i\}$. Un sólo plásmido es necesario para representar un individuo del espacio de búsqueda del problema, el cual es introducido en la computación por medio de las bacterias de inicialización⁵ B_0 .

El plásmido tiene $n + 1$ casetes característicos. El primero de todos ellos se usa para codificar un promotor constitutivo⁶ y los n siguientes para codificar los valores positivos de las variables del enunciado (x_1, x_2, \dots, x_n) . Esta forma de codificar el conjunto de datos solución es altamente susceptible a cambios, por ejemplo se podría colocar un promotor constitutivo delante de cada casete o variable.

Los sistemas de intercambio de casetes de cada bacteria se encargan de cambiar las variables -inicialmente positivas- de este conjunto de datos por variables del

⁵Pertenecientes a la cepa *extra*, sin instrucciones cromosómicas, encargada de transportar los conjuntos de datos.

⁶Por ejemplo, un promotor de Neomicina, pNm .

conjunto (x_1, x_2, \dots, x_n) formando así individuos nuevos.

Las variables del enunciado del problema se codifican cada una en un gen estructural cuya expresión genética produce una molécula específica. Así, las moléculas expresadas, correspondientes a un conjunto de datos, pueden interactuar con la instrucción de evaluación insertada en el cromosoma bacteriano y producir, si el plásmido codifica el conjunto solución, una salida positiva al problema.

Recombinación dirigida. La recombinación es una útil y maleable operación que las bacterias llevan a cabo frecuentemente con la finalidad de mutar o cruzar información genética [Fra07]. Así, se puede modificar este comportamiento para satisfacer las necesidades de la computación que queremos que las células realicen. En este caso, los procesos de recombinación dirigida tienen como objetivo el intercambio de casetes entre el plásmido y el cromosoma bacteriano de manera que la información final que se encuentra en el conjunto de datos -o plásmido- después de este proceso corresponda a un individuo nuevo del espacio de búsqueda del problema.

Para realizar este intercambio de información genética se hace uso de los sistemas de intercambio de casetes por medio de proteínas recombinasas (RMCE⁷) Estos sistemas son cada día más usados en biología sintética así como en ingeniería genética debido a la utilidad de éstos para manipular información genética [Bae01].

Los sistemas RMCE se definen por el par proteína-sitio en el que una proteína recombinasa es capaz de intercambiar dos casetes que estén flanqueados por sitios de recombinación específica. Aunque la variedad de sistemas RMCE no es muy extensa, se consigue manipular los sitios específicos para crear múltiples subsistemas mutados de un sistema dado.

El método de realizar procesos de recombinación dirigida es diseñado para evitar la generación de espacio de búsqueda no útil y reducir así el coste computacional del algoritmo. En el problema SAT, un individuo válido del espacio de búsqueda del problema es aquel conjunto de datos que presenta una computación final (ya resulte

⁷Recombinase-Mediated Cassette Exchange.

True o *False* su validación). Es decir, debe presentar valores positivos o negativos para las n variables del enunciado.

Todos aquellos individuos que contienen un número de valores diferente al número de variables son obviamente inválidos. También lo son, aquellos que teniendo n valores, varios de ellos hacen referencia a la misma variable de la fórmula enunciado dejándo sin valor definido a algún parámetro. Estos individuos inválidos no producen en el proceso de validación una computación final y son considerados pertenecientes al espacio de búsqueda inútil.

El proceso de recombinación dirigida optimiza este cruzamiento y asegura que sólo los individuos válidos son generados durante el proceso evitando así búsquedas sin sentido.

En la comunidad de bacterias habrá, por tanto, tantas instrucciones de recombinación como variables tenga la fórmula enunciado. Cada cepa bacteriana estará genéticamente modificada para llevar a cabo una recombinación dirigida específica en su interior, encontrándose en un número igual al número de parámetros inicial.

Instrucción de evaluación. El problema SAT, tal como aquí se plantea, necesita una evaluación externa separada de los datos que compruebe la viabilidad de los valores asignados a las variables. La idea de tener almacenadas las instrucciones de evaluación como *programas* fijos en los cromosomas capaces de evaluar los conjuntos de datos presentes en una célula, es innovativa y útil a la hora de traducir los clásicos experimentos *in-vitro* a un nuevo paradigma autónomo *in-vivo*.

Cada año decrece exponencialmente el coste de síntesis de ADN y, algún día, será posible sintetizar el genoma bacteriano completo. El proceso de almacenaje de instrucciones cromosómicas como se propone en este capítulo podría resultar entonces de gran utilidad y potencia de cómputo en el diseño de biodispositivos celulares.

Esta instrucción de evaluación se comporta como un circuito genético simple en el que las entradas son los factores de transcripción resultantes de la expresión

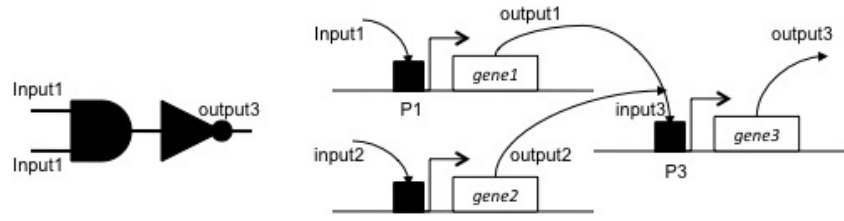


Figura 9.2: Instrucción de evaluación. Equivalencia entre circuitos lógicos (izquierda) y genéticos (derecha).

del conjunto de datos. Por ejemplo, la figura 9.2 muestra el circuito genético que sirve como instrucción de evaluación correspondiente a la fórmula $\neg(a \wedge b)$. Al estar la fórmula formada únicamente por un literal, la ecuación SAT se puede traducir en una puerta lógica clásica, siendo la puerta NAND la que presenta esa funcionalidad.

Los promotores $P1$ y $P2$ de la figura 9.2 actúan como promotores inducibles por la presencia de los factores de transcripción de las variables a y b respectivamente. Por el contrario, el promotor $P3$ es un promotor represible por el factor de transcripción conjunto que surge de la unión de los parámetros $output1$ y $output2$. De esta forma, cuando los promotores inducibles expresan sus genes consecuentes el promotor $P3$ resulta reprimido y el gen $gene3$ no se expresa. En cualquier otro caso, este gen final sí es expresado cumpliendo así esta instrucción con la fórmula lógica propuesta.

9.4. Computando una instancia del SAT

Para finalizar este capítulo, se formaliza una simulación teórica de una instancia del problema SAT formada por dos literales y dos cláusulas.

La expresión lógica del problema que debe ser resuelto de manera autónoma por una comunidad multi-cepa transgénicamente modificada de la manera detallada previamente es:

$$(\neg a \wedge \neg b) \quad (9.7)$$

El espacio de búsqueda correspondiente a la ecuación anterior está formado

por las 4 posibles soluciones a ésta:

$$[\{a, b\}, \{\neg a, b\}, \{a, \neg b\}, \{\neg a, \neg b\}] \quad (9.8)$$

A partir del conjunto de datos dado al principio de la computación e introducido en la comunidad *armada*⁸ por cada bacteria inicial B_0 , el resto del espacio de búsqueda se va explorando de manera autónoma por la comunidad y se evalúan las soluciones encontradas a mismo tiempo que se forman éstas. Este conjunto inicial de datos corresponde al plásmido que codifica el individuo $(a \wedge b)$, el cual presentará resultado negativo en el proceso de evaluación al no representar una solución válida de la ecuación .

Son tres las cepas necesarias para formar la comunidad multi-cepa capaz de procesar la instancia SAT propuesta. Ya que la ecuación del enunciado está compuesta por dos variables se necesitarán dos cepas principales modificadas para integrar instrucciones de recombinación en sus respectivos cromosomas (un sistema RMCE distinto en cada cepa), y otra cepa extra encargada de portar las copias del plásmido que representa el conjunto de datos inicial.

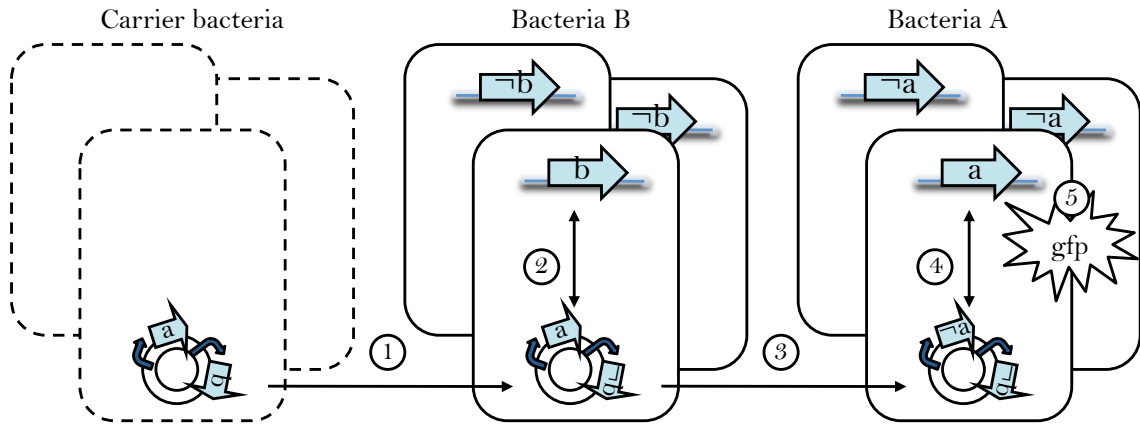


Figura 9.3: Algoritmo de resolución de la ecuación propuesta en una comunidad formada por tres cepas: una portadora (*carrier bacteria*) y dos principales.

La figura 9.3 muestra el comportamiento de la comunidad bacteriana desde un

⁸Recordar la diferencia entre biodispositivo armado y biodispositivo activado explicada anteriormente en este capítulo.

punto de vista de la computación a alto nivel. La secuencia de acciones ejemplificada en la figura se detalla a continuación:

1. El plásmido con el conjunto de datos inicial ($a \wedge b$) se introduce en la comunidad armada compuesta por las dos cepas principales por medio de la cepa portadora. A continuación, cualquier bacteria de dicha cepa puede realizar un proceso de transferencia horizontal con otra bacteria receptora. En el ejemplo de la figura 9.3 se produce una conjugación entre una bacteria inicial y una bacteria de la cepa B (cepa específica de la segunda variable).
2. Una vez el plásmido ha sido transferido, el sistema de recombinación Cre/lox (sistema RMCE de la cepa B como se detallará a continuación) modifica el plásmido sustituyendo la variable b por su valor negativo $\neg b$ que inicialmente posee la bacteria en el cromosoma. El conjunto de datos representa ahora la solución ($a \wedge \neg b$) la cual no valida la ecuación 9.4 y en consecuencia la bacteria no mostrará salida positiva.
3. Otro proceso de conjugación bacteriana se produce entre la última bacteria B y una bacteria A receptora.
4. Una vez transferido el plásmido (que ya no representa el conjunto de datos inicial) el sistema de recombinación FLP/FRT (sistema RMCE de la cepa A) intercambia el casete a del plásmido por el casete $\neg a$ del cromosoma bacteriano.
5. La nueva configuración del conjunto de datos ($\neg a \wedge \neg b$) representa los valores correctos que validan la ecuación 9.4. La expresión de los dos genes correspondientes a dichas variables provoca que la instrucción de evaluación resulte en la expresión de un *reporter* como un gen de fluorescencia.

Como se ha mencionado, son dos los sistemas de recombinación dirigida los que se utilizan en el diseño de este biodispositivo:

1. FLP/FRT. Este sistema RMCE lo utilizan las bacterias de la cepa A y mediante la producción de la proteína recombinasa FLP pueden intercambiar

los casetes flanqueados por sitios FRT. Se utiliza un sitio FRT y otro FRT^m (mutado) para rodear los genes objetivo.

2. Cre/lox. Por medio de la proteína Cre las bacterias de la cepa B intercambian los casetes flanqueados por sitios de recombinación específicos lox. Al igual que el caso anterior, se utilizan dos tipos diferentes de sitios lox: el sitio loxP y un sitio lox^m mutado como lox2272.

La instrucción de evaluación codificada en los cromosomas de las cepas principales consiste en un único promotor con comportamiento AND (promotor T7). Este promotor es inducido por la presencia de dos factores simultáneamente que son: T7 RNA polimerasa -expresada mediante el gen estructural *T7ptag*- y un supresor tRNA expresado por el gen *supD* [And07].

La figura 9.4 muestra en detalle la codificación de las estructuras de información manejadas por el biodispositivo en el que se utilizan las correspondencias de la tabla 9.1.

Cuadro 9.1: Correspondencia genética de las variables del enunciado.

Variable	Gen correspondiente
a	araC (Arabinose)
b	IPTG (Isopropyl-beta-bio D-galactopyranoside)
$\neg a$	supD (Amber suppresor tRNA)
$\neg b$	T7ptag (T7 RNA polymerase)

En la figura 9.4a se detalla la codificación del plásmido inicial con la configuración $(a \wedge b)$ y los cromosomas (antes de cualquier conjugación) de las cepas principales A y B. El cuarto paso de la figura 9.3 se muestra especificado en la figura 9.4b en el que el plásmido con la configuración $(a \wedge \neg b)$ se encuentra en el interior de una bacteria de la cepa B. En el momento de la conjugación se produce la proteína FLP que permite el intercambio entre el primer casete característico del plásmido (rodeado de sitios específicos FRT) y el gen supD almacenado inicialmente en el cromosoma bacteriano.

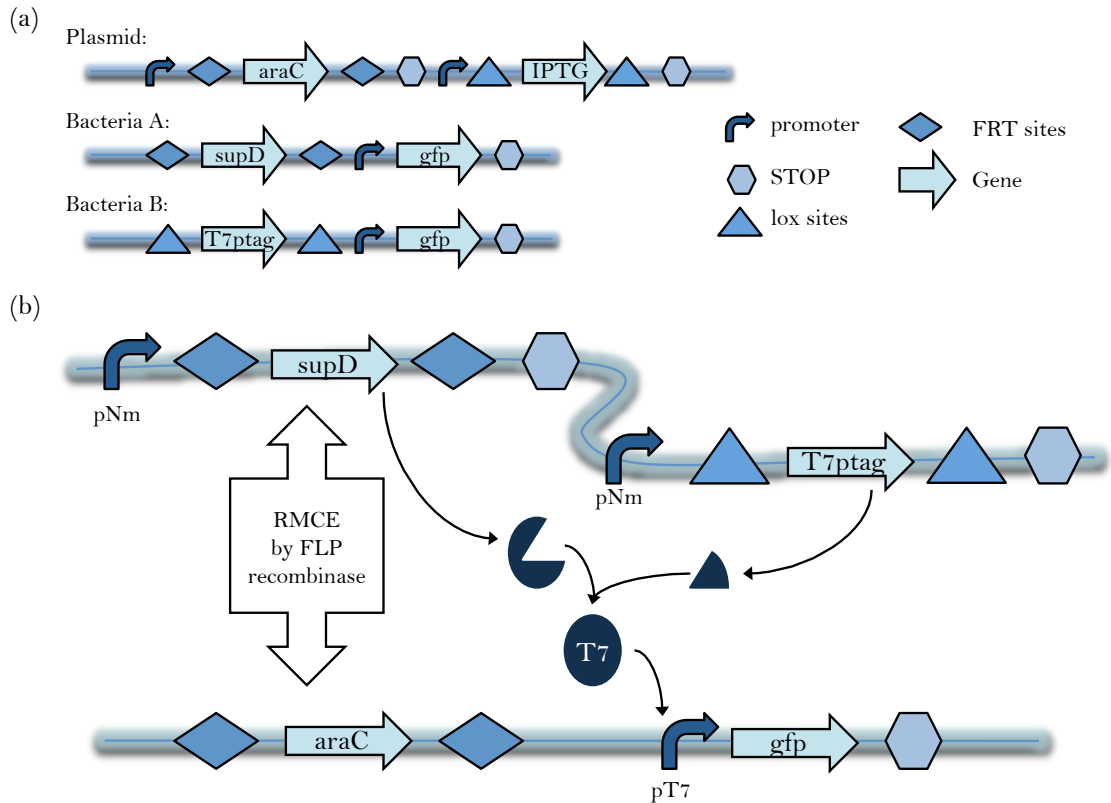


Figura 9.4: Codificación de la información del problema. a) Codificación del conjunto de datos inicial así como la configuración de las dos cepas principales. b) Ejemplo de intercambio de casetes producido en el interior de una bacteria de la cepa A.

Como resultado, ambos genes del plásmido al expresarse forman el factor de transcripción T7 el cual induce el promotor pT7. Se produce así la expresión del *reporter* que en este caso es un gen de fluorescencia verde (*gfp*) identificando así aquella bacteria que ha llegado a obtener a la solución del problema.

9.4.1. Discusión

Es importante resaltar el carácter totalmente asíncrono que presenta la computación en el biodispositivo. Es decir, los ciclos de recombinación no se suceden secuencialmente sino paralelamente de un modo aleatorio, pudiendo coexistir conjuntos de datos muy evolucionados con conjuntos de datos aún en su fase inicial. Los flujos de datos observados en la figura 9.5 ilustran el camino por el que los conjuntos de

datos transitan de manera asíncrona en una comunidad que resuelva una instancia con cuatro variables.

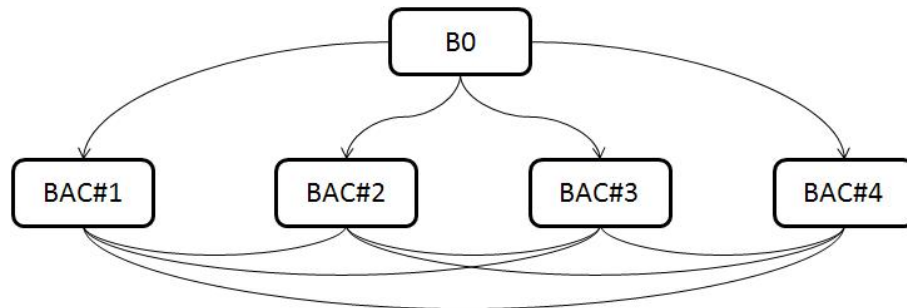


Figura 9.5: Flujo de datos en la comunidad SAT.

El paralelismo masivo de los procesos de conjugación entre bacterias⁹ y el carácter autónomo que presenta la comunidad heterogénea a la hora de resolver el problema planteado son los objetivos perseguidos por este desarrollo.

También es importante resaltar que el proceso de apareamiento entre células beneficia el buen desarrollo del algoritmo planteado al no permitir que una bacteria funcione como receptora de plásmidos más de una vez siendo así identificable el conjunto de datos que contiene. Esta secuencia es:

- Primer paso. Una célula conjugativa (F+), con su pelo sexual se acerca a otra no conjugativa (F-)¹⁰.
- Segundo paso. Contacto entre las dos células por medio del pelo sexual. Se produce entonces un acercamiento de ambas hasta entrar en contacto, formándose entonces en las membranas un poro por el que se transmite el material genético.
- Tercer paso. Síntesis de las cadenas de ADN conjugativo y formación del plásmido de doble cadena.

⁹La concentración de bacterias por mililitro de solución supera los 2×10^9 células.

¹⁰El carácter conjugativo de las bacterias va implícito en el plásmido de datos inicial. Por tanto, sólo las bacterias de la cepa inicial son, al principio de la computación, bacterias conjugativas con factor F+.

- Cuarto paso. Sellado de los poros y separación de las células. Cada una de ellas contiene ahora una copia del Factor F.

Al acabar el cuarto paso de la conjugación, la bacteria F- se convierte en F+ lo que la confiere una importante característica: a partir de ese momento, la bacteria no puede ser receptora nunca más de plásmidos y si donante. Esta faceta hace que los conjuntos de datos no acaben mezclándose de manera inproductiva y el algoritmo se desarrolle sin problemas.

La finalidad de este paradigma de cómputo por conjugación, más allá de resolver un problema matemático, es mostrar la capacidad de una comunidad de bacterias de evolucionar un conjunto de información genética hacia un determinado objetivo mediante la utilización de las rutas de comunicación naturales. Esta mutación dirigida por instrucciones genéticas puede resultar de utilidad en aplicaciones ecológicas o médicas en donde el carácter autónomo del biodispositivo representa una gran ventaja.

Capítulo 10

Diseño de un biocircuito celular

Índice

10.1. Arquitectura del Modelo	106
10.2. Elementos estructurales: puertas lógicas celulares	108
10.3. Modelado del sistema	113
10.4. Simulación	119

En este capítulo se propone el diseño detallado de un circuito genético lógico utilizando una comunidad heterogénea de bacterias formada por tres cepas transgénicas independientes. Cada cepa presenta una funcionalidad específica de manera que sólo mediante la unión de las tres, el comportamiento final del circuito pueda emerger. Este comportamiento final responde a una función lógica XOR.

10.1. Arquitectura del Modelo

La alteración de los sistemas de comunicación bacteriana interespecie (*quorum sensing*) basada en el intercambio de pequeñas moléculas de la familia N-Acylhomoserine lactone (AHL) sirve para realizar las tareas de sincronización necesarias en el circuito.

Como ya se ha visto, los circuitos genéticos son mayoritariamente modelados

en una sola célula desarrollándose así la funcionalidad completa del circuito en el interior de la misma. Sin embargo, en este capítulo se propone un diseño en el que cada célula B es considerada una puerta lógica del circuito total y las conexiones químicas célula-célula son modificadas para controlar la funcionalidad final que se desarrolla en la comunidad multi-cepa en su conjunto. Siendo una cepa C representada por un conjunto de bacterias $\{B_1, B_2, \dots, B_n\}$ transgénicamente modificadas por igual para presentar una funcionalidad lógica determinada, un biocircuito A estará formado por un conjunto de cepas, $A = \{C_1, C_2, \dots, C_m\}$ donde m representa el número de puertas lógicas necesarias para la correcta ejecución del circuito lógico diseñado.

Una de las mayores ventajas que este paradigma presenta a simple vista es el amplio carácter reutilizable resultante de tener una cepa genéticamente modificada por cada puerta lógica. Esto permite el fácil desarrollo de biocircuitos con comportamientos diferentes con el simple hecho de combinar las cepas de distintas formas en lugar de tener que reprogramar toda la red genética intracelular, como se tendría que hacer en los desarrollos uni-cepa.

De esta manera se presenta un modelo de biocircuito lógico XOR (*exclusive OR*) formado por tres cepas bacterianas: una cepa modificada para que cada una de sus bacterias funcione como una puerta OR, otra formada por bacterias con funcionalidad NAND y una última cuyas células muestran un comportamiento AND.

Este diseño responde al funcionamiento de una arquitectura de comunicaciones básica en la que la comunicación entre cepas fluye únicamente en la dirección *capa de entrada* \rightarrow *capa de salida* y la capa de entrada debe ser estimulada desde el exterior de la población cada vez que se desea obtener una respuesta. Así, el biocircuito se comporta como un circuito digital convencional en el que no existe retroalimentación alguna y sólo la presencia de entradas provoca las reacciones necesarias para la obtención de una salida.

En el capítulo siguiente se propone la ampliación de esta arquitectura de comunicaciones mediante el uso de señales de *feedback* entre las capas de la comunidad multi-cepa.

Para el caso del biocircuito, que es el objetivo de este capítulo, la distribución de cepas en las capas se realiza de la siguiente manera:

- Capa de entrada: compuesta por dos cepas, con funcionalidades OR y NAND respectivamente, encargadas de recibir las dos señales de entrada del biocircuito.
- Capa de salida: compuesta por una cepa, de comportamiento AND, diseñada para recibir la señalización proveniente de la capa de entrada y actuar en consecuencia para devolver la salida que corresponde al comportamiento XOR en forma de intensidad lumínica.

Como se apuntaba anteriormente, la sincronización entre las distintas capas del biodispositivo se realiza mediante el intercambio de moléculas AHL. Estas moléculas son sintetizadas en la bacteria origen por el gen LuxI y reconocidas en la bacteria destino por el gen LuxR. Si se realizan pequeñas alteraciones en la secuencia de estos genes se pueden provocar cambios en las moléculas AHL de manera que pueden haber tantas señales de sincronización distintas como sea necesario. El biocircuito XOR detallado en la sección de modelado hace uso de dos pares LuxI-LuxR diferentes.

10.2. Elementos estructurales: puertas lógicas celulares

Para poder construir el nanodispositivo correspondiente a un biocircuito capaz de desempeñar una funcionalidad lógica específica mediante el uso de bacterias, es importante definir previamente y como punto de inicio los elementos estructurales útiles para estos desarrollos. Estos elementos son las puertas lógicas necesarias para el diseño de circuitos complejos las cuales estarán contenidas cada una en una bacteria distinta como se ha comentado previamente.

Las células lógicas o puertas celulares, como se llamarán de aquí en adelante, deben ser dispositivos capaces de procesar una o más entradas y expresar una salida

como consecuencia de la naturaleza de las señales de entrada siguiendo la función lógica para la cual han sido genéticamente modificadas.

El diseño de células lógicas está basado en la capacidad de alterar las rutas y mecanismos naturales de transcripción para forzarlas a representar funcionalidades lógicas. Esta alteración consiste en incluir en el cromosoma de las bacterias una determinada secuencia que marcará su función específica como las detalladas más adelante en el caso de las bacterias lógicas AND, OR y NAND que son las que intervienen en el circuito XOR objetivo de este capítulo. Se diseña entonces una puerta lógica por cepa bacteriana ya que todos los individuos que forman una misma cepa son genéticamente modificados de la misma forma y, por tanto, tienen la misma secuencia alterada en el cromosoma.

Gracias a los importantes avances en tecnología biomolecular, la inserción de secuencias en el cromosoma bacteriano es cada vez una tarea más sencilla y fiable. Hoy en día, los esfuerzos en la investigación relativa a la secuenciación del genoma bacteriano y la catalogación y anotación de las unidades funcionales del mismo, revelan la existencia de unas determinadas secuencias denominadas *neutral sites* o espacios neutrales en los cuales se pueden insertar nuevas unidades sin peligro de afectar a la funcionalidad vital de la célula o su viabilidad. Son esos espacios los utilizados para insertar las unidades funcionales lógicas.

Una gran diversidad de operaciones pueden llevarse a cabo utilizando puertas celulares debido a la naturaleza analógica con la que se trabaja a nivel molecular. Las bacterias no están obligadas o restringidas, como lo están los computadores convencionales, a trabajar únicamente de manera digital por lo que los dos paradigmas computacionales, digital y analógico se encuentran intrínsecos en los procesos celulares. Es por ello, que futuros desarrollos referidos a circuitos genéticos deben explotar más la vertiente analógica en detrimento de la digital ya que, combinando ambas, se podrían diseñar dispositivos de características inalcanzables por un solo paradigma.

En la figura 10.1 se muestra, de manera parcial, el catálogo de bacterias lógicas (digitales y analógicas) que se pueden diseñar utilizando las células como unidades

independientes de proceso y que sirven para su uso como elementos estructurales en el diseño de biocircuitos basados en poblaciones multi-cepa.

Tanto la parte de puertas lógicas digitales o puertas lógicas estrictas de la figura 10.1 como la parte correspondiente a puertas lógicas analógicas de dicha figura, detallan el funcionamiento de varias posibilidades en el comportamiento de puertas bacterianas desde el punto de vista de un diseño de alto nivel. Esto es así ya que no se detalla la composición interna de la bacteria ni la secuencia genómica específica que determina su comportamiento lógico, sino que únicamente se observa el comportamiento final de éstas. Las moléculas que se muestran en la figura representando las entradas y/o salidas de las puertas celulares tienen la naturaleza siguiente:

- Moléculas $\{a, b, c\}$: son señales de entrada de la familia AHL¹ que inducen la expresión genética en el interior de las puertas celulares.
- Moléculas $\{d\}$: son señales de salida que representan o bien una molécula perteneciente al conjunto $\{a, b, c\}$ o bien una proteína indicadora o marcadora. Ejemplos de estas proteínas son la Luciferasa y la proteína GFP.
- Moléculas $\{e\}$: son únicamente proteínas indicadoras o marcadoras usadas en respuestas finales.

La figura 10.1.1) representa una implicación primaria en la que una molécula externa provoca en la bacteria la activación de su funcionalidad y la consecuente producción de otra molécula de diferente o igual naturaleza. Las figuras 10.1.2) y 10.1.3) ilustran el comportamiento de puertas celulares AND. Las secuencias lógicas de los cromosomas de estas bacterias están diseñadas de manera que la bacteria producirá moléculas de respuesta sólo si las dos entradas están presentes en el mismo instante de tiempo. Es importante hacer notar la diferencia entre las dos figuras: el carácter interno (se encuentra en el interior de la membrana celular) o externo (está presente en la solución nutriente) de las entradas. El objetivo de usar como entrada una molécula del interior de la célula es el diseño de biocircuitos más complejos en los que la comunicación célula-célula coopere con los circuitos regulatorios

¹Acylhomoserine lactone

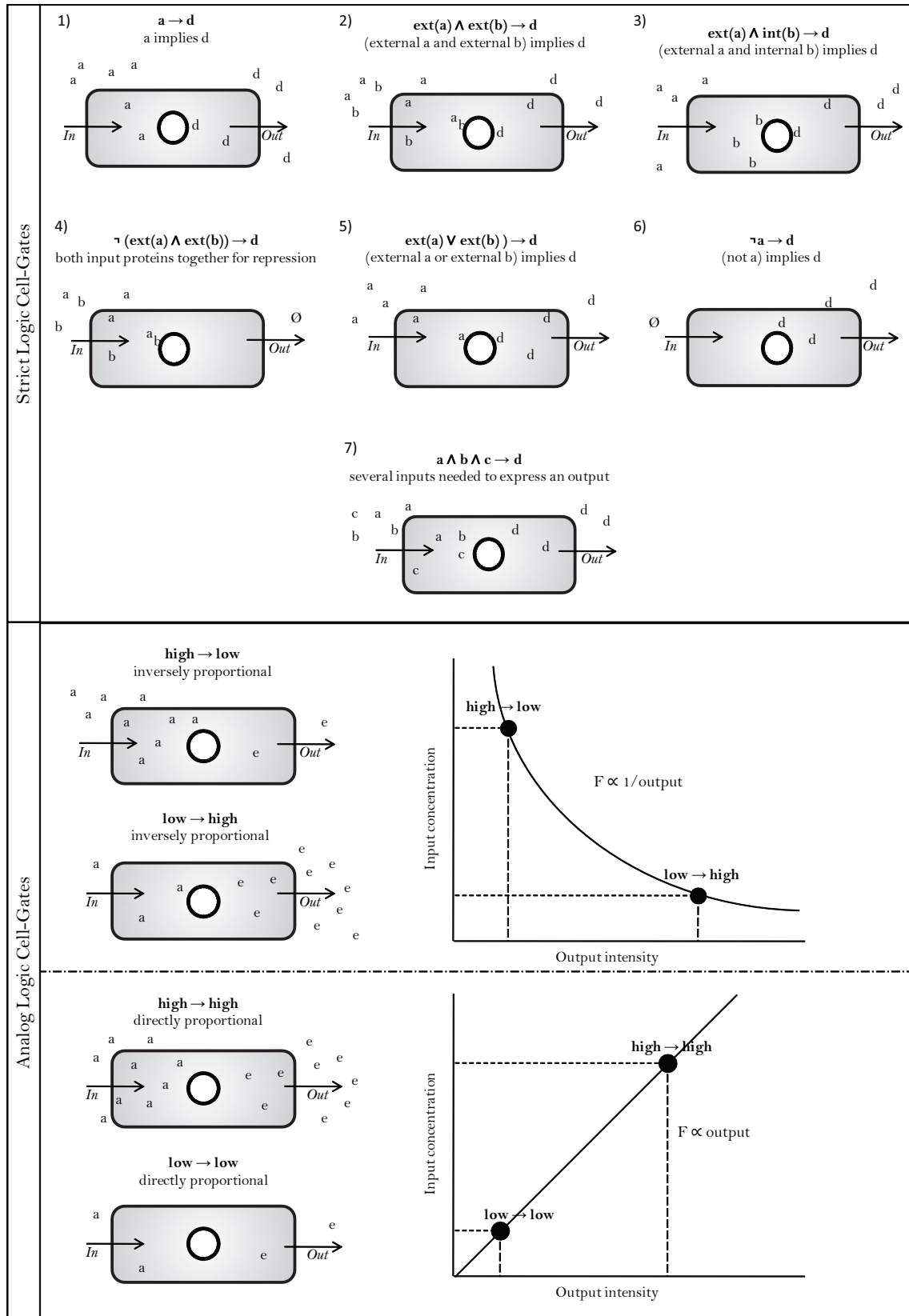


Figura 10.1: Puertas lógicas celulares

intracelulares. Esta característica no es usada en el modelado del biocircuito XOR, pero se propone como línea de investigación futura.

La bacteria de la figura 10.1.4) se comporta como una puerta NAND, es decir, si ambas moléculas de entrada están presentes la bacteria no producirá ninguna respuesta.

La siguiente bacteria, 10.1.5), presenta funcionalidad de puerta OR en la que la sola presencia de una de las entradas provoca una respuesta celular y la bacteria 10.1.6) es una puerta celular inversora. Esta última devuelve moléculas de respuesta cuando la entrada no está presente y viceversa.

Con el diseño de la puerta celular 10.1.7) se quiere representar la posibilidad de utilizar puertas lógicas estrictas multi-entrada. En este caso, la puerta se comporta como una función AND triple, es decir, se necesita que las tres entradas esten presentes en el mismo instante de tiempo para poder expresar una respuesta final. Esta última posibilidad muestra el carácter altamente escalable de este paradigma de cómputo al poder *programar* varias funciones lógicas en una misma puerta celular.

En el apartado donde se detallan las puertas analógicas de la figura 10.1, se pueden observar dos gráficos que muestran la relación entre el nivel de señal de entrada y la intensidad del marcador de salida (intensidad lumínica en el caso de usar Luciferasa o GFP). La primera puerta celular representa una función inversamente proporcional entre entrada y salida: a mayor valor de entrada menor valor se obtiene en la salida. De manera contraria, la segunda puerta celular se comporta siguiendo una función directamente proporcional. Estas puertas codifican un comportamiento analógico real y pueden resultar muy útiles en la construcción y diseño de circuitos genéticos complejos.

En [Nor08] se propone, aunque de una manera superficial, una posible solución a una instancia del problema TSP mediante el uso de bacterias con este comportamiento. Así las entradas corresponderían a la distancia y la salida a la evaluación de estas, de manera que a mayor distancia de un camino concreto, menor sería la respuesta de la bacteria que lo evalúa.

10.3. Modelado del sistema

De forma paralela a un circuito lógico digital, que es un dispositivo hardware formado por puertas lógicas situadas sobre una superficie rígida, el biocircuito modelado consta de puertas celulares capaces de procesar datos e información lógica contenidas en una solución nutriente.

En el modelo descrito, las puertas lógicas están implementadas utilizando una cepa genéticamente modificada por cada función lógica con el motivo de resaltar la potencia computacional de la comunicación intercelular, el paralelismo y la rehusabilidad.

Dos importantes ventajas surgen de la forma en que se diseñan y sincronizan los flujos de información entre la entrada/salida de una bacteria y la entrada/salida de siguiente:

- Señales concurrentes.

Circuitos cuyas puertas están todas completamente interconectadas surgen como resultado de mezclar en el mismo soporte varias cepas de puertas celulares. Así, en un biocircuito C que contiene n bacterias, $C = [B_0..B_n]$, la salida de una de esas puertas B_i es evaluada por el resto de la población de forma paralela:

$$B_{i_out} \rightarrow B_{n_in} \forall n \in C \quad (10.1)$$

- Umbrales de sincronización.

Con la finalidad de sincronizar toda la población para ofrecer una respuesta simultánea de las puertas celulares, y como aproximación al control del ruido genético, las bacterias utilizan umbrales regulatorios de la señal de entrada. Cada una de las puertas B_i de la población tiene un umbral δ asignado. El parámetro δ indica el nivel de concentración mínimo de moléculas de entrada necesarias para producir una respuesta en la célula objetivo. Si la salida producida por una bacteria B_i debe ser usada como la entrada en la bacteria B_j

y el número de moléculas de señalización se denota por $|x|$, se puede afirmar que cuando se cumple la ecuación:

$$|B_{i_out}| \geq \delta_{B_j} \quad (10.2)$$

la función lógica de B_j es activada.

La comunicación entre las puertas tiene lugar en el agente intermedio que hace el papel de *buffer* o memoria compartida. La solución nutriente mineral necesaria para cultivar la población de bacterias en un entorno que facilite sus acciones metabólicas realiza las funciones de esa memoria, a través de la cual las señales se comparten y transmiten.

En el momento en el que las bacterias son situadas en este medio, se comportan como procesadores independientes que realizan permanentemente procesos de lectura por medio de su membrana. El objetivo de esta lectura del medio es encontrar las moléculas apropiadas para activar la función lógica para la cual la bacteria ha sido previamente diseñada.

Del mismo modo las células realizan los procesos de escritura en la memoria compartida correspondientes a su función lógica con la finalidad de transmitir señales a la siguiente puerta del circuito. El paralelismo masivo en las conexiones viene dado por el hecho de que en un mililitro de solución nutriente se encuentran aproximadamente 2×10^9 puertas celulares, las cuales llevan a cabo simultaneamente procesos de lectura/escritura.

La función lógica XOR, o puerta OR exclusiva, entre dos señales A y B se define como aquella que da resultado positivo cuando una de sus señales de entrada es positiva, pero no las dos al mismo tiempo tal como se muestra en la ecuacion 10.3. Por tanto, la respuesta de la puerta celular debe responder a la siguiente afirmación: NOT A y B, o A y NOT B.

$$A \oplus B = \overline{A}B + A\overline{B} \quad (10.3)$$

La figura 10.2 muestra la traducción del circuito lógico esquemático XOR (10.2.1) en el biodispositivo *in-vivo* de función análoga implementado con puertas celulares (10.2.2). Como ya se ha mencionado previamente, este biocircuito consta de tres cepas transgénicas diferentes: OR (10.2.2a), NAND 10.2.2b) y AND (10.2.2c).

Es importante hacer hincapié en el carácter de máquina ensamblada que tiene la comunidad XOR. Esto es así ya que el biodispositivo final sólo presenta la funcionalidad requerida cuando todos sus componentes se juntan para tal fin. Es decir, estando la población XOR formada por tres cepas independientes, un subconjunto de dichas cepas no presentaría ninguna funcionalidad siendo necesaria la intervención de todos los componentes de la máquina para que ésta quede armada y activada simultáneamente.

En cuanto a la filosofía de activación, este biodispositivo es ligeramente distinto al diseño del capítulo anterior aunque mantiene la separación entre los conceptos de biodispositivo armado y biodispositivo activado. Aquí, el estado armado corresponde a la mezcla en el mismo cultivo de las tres cepas mientras que el estado de activación se alcanza una vez introducidas las moléculas de entrada en la comunidad.

Al producir el biocircuito salidas que son directa y estrictamente dependientes de las entradas, se trata de modelar un circuito lógico combinatorio en el que los elementos estructurales se ven afectados únicamente por las entradas del circuito y las salidas parciales de las puertas precedentes en lugar de verse afectados en instantes discretos de tiempo por pulsos de reloj.

Analizando el circuito desde el punto de vista del diseño de alto nivel de la figura 10.2.1 se pueden observar cinco señales de entrada/salida. Las moléculas de la clase Arabinosa² e IPTG³ actúan como las señales de entrada al biocircuito que deben iniciar la cascada de acciones necesaria para concluir en una salida lumínica. Dichas moléculas son leídas por las cepas OR y NAND, las cuales dependiendo de la presencia o ausencia las entradas actuarán de manera diferente. Las respuestas

²Monosacarido ampliamente usado en cultivos bacterianos compuesto por cinco carbonos y un grupo aldehído.

³Abreviatura de Isopropyl β thiogalactopyranoside.

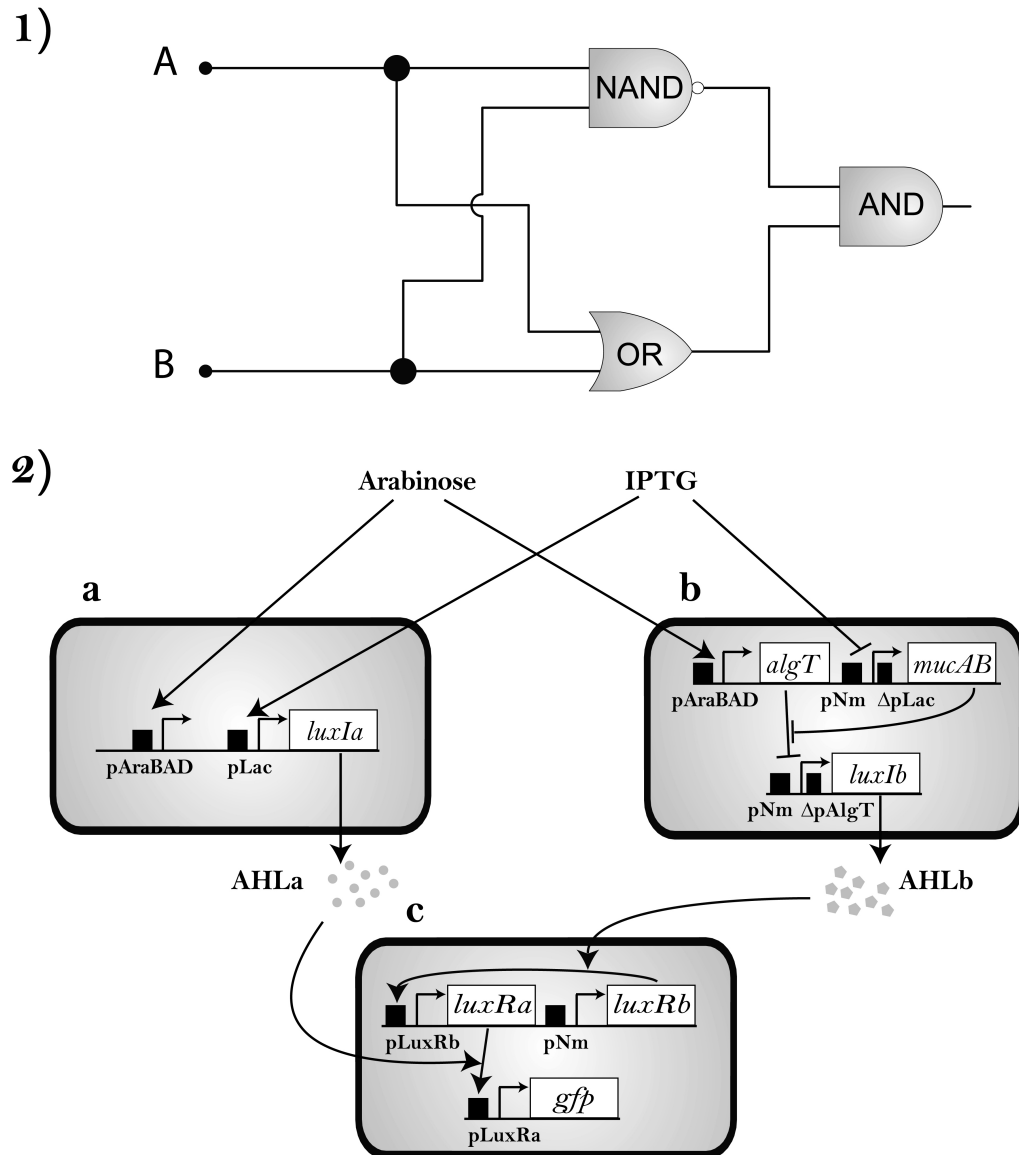


Figura 10.2: Diseño de un biocircuito XOR. 1) Representación esquemática de un circuito XOR digital. 2) Diseño biológico del circuito XOR usando 3 cepas transgénicas; 2a: puerta celular OR; 2b: puerta celular NAND; 2c: puerta celular AND. *pAraBAD*: promotor inducible por Arabinosa; *pLac*: promotor inducible por IPTG; $\Delta pLac$: promotor inducible truncado IPTG; $\Delta pAlgT$: promotor truncado *algT*; *pNm*: promotor constitutivo de Neomicina; *gfp*: proteína de fluorescencia verde.

Cuadro 10.1: Reglas de inferencia de alto nivel

R1	if $Arabinosa \vee IPTG$ then $OR(a) \longrightarrow AHL_a$
R2	if $\neg Arabinosa \vee \neg IPTG$ then $NAND(b) \longrightarrow AHL_b$
R3	if $(AHL_b \geq \delta_c) \wedge (AHL_a \geq \delta_c)$ then $AND(c) \longrightarrow gfp$

de estas dos cepas corresponden a la producción o no de moléculas del tipo AHL_i , donde $i = \{a, b\}$, que son escritas en la memoria compartida (esparcidas por la solución nutriente) de manera que la cepa correspondiente a la función lógica AND pueda leer el medio en busca de éstas. Consecuentemente y siguiendo la instrucción lógica incluida en el cromosoma de la cepa AND, estas bacterias producirán o no la proteína de fluorescencia verde.

Mediante esta luz fácilmente detectable en laboratorio⁴, el biocircuito alcanzará la solución final de la computación de tal manera que la presencia de luz significará la salida positiva de ésta.

Las tres reglas de inferencia que explican el comportamiento de la población bacteriana heterogénea a alto nivel se enumeran en la tabla 11.1 en forma de relaciones sintácticas entre premisas y conclusiones. La primera regla R1 hace referencia a la puerta celular OR y en ella se establece la producción de AHL_a siempre que *Arabinosa* o *IPTG* estén presentes. La segunda, R2, detalla el comportamiento de la bacteria NAND, para la cual se condiciona la producción de AHL_b a la ausencia de cualquiera de sus entradas. La última regla, R3, corresponde a la bacteria perteneciente a la capa de salida del biodispositivo (AND) e indica la necesidad de recibir los dos tipos de moléculas AHL para producir la proteína fluorescente *gfp*.

En el diseño de bajo nivel de las bacterias lógicas ilustrado en la figura 10.2.2 se han utilizado dos tipos distintos de promotores en lo que se refiere al comportamiento de éstos: promotores constitutivos que permiten la expresión genética constantemente y promotores inducibles, los cuales necesitan activación previa para inducir la expresión.

⁴Mediante unos dispositivos llamados fotomultiplicadores.

El funcionamiento se detalla a continuación:

Cepa bacteriana OR. Los individuos pertenecientes a esta cepa expresan *luxIa* siempre que un promotor inducible esté siendo activado. Esto ocurre cuando el componente Arabinosa induce pAraBAD o cuando IPTG induce el promotor pLac.

Cepa bacteriana NAND. Estas puertas celulares expresan *luxIb* siempre y cuando ambas entradas no estén presentes al mismo tiempo en la comunidad. El aspecto más relevante de este diseño se centra en la regulación de la actividad de los promotores de Neomicina (pNm). Estos promotores inducen constantemente la expresión de los genes situados a continuación como *mucAB* y *luxIb* a no ser que algún factor de transcripción se ligue a los promotores truncados que los acompañan. La otra característica fundamental es que la expresión constante del sistema MucAB bloquea postranscripcionalmente los factores de transcripción AlgT [Mat97]. Así, estas puertas celulares paran la expresión de *luxIb* y, por consiguiente, la producción de AHL_b si y solo si ambas entradas, Arabinosa e IPTG, son entradas verdaderas. IPTG desactiva el complejo pNmΔpLac, bloqueando la producción de *mucAB* y Arabinosa induce AlgT reprimiendo así al complejo pNmΔpAlgT.

La tabla 11.2 hace referencia a las reglas de inferencia del diseño de bajo nivel. Para comprender la secuencia de reglas que finaliza en expresión lumínica para un caso inicial en el que se introduce únicamente Arabinosa (y no IPTG) en la solución, conviene determinar previamente el estado inicial del dispositivo. Todos los promotores, excepto los promotores de Neomicina constitutivos, se encuentran en estado INACTIVO ya que son inducibles y necesitan activación mediante factores de transcripción externos. Por lo tanto, todos los genes excepto *mucAB*, *luxIb* y *luxRb* se encuentran en estado FALSE.

Formalmente se dice que, siendo L_i el estado de la bacteria B_i en un instante específico, la configuración $I_1 = (L_1, L_2, L_3)$ cambia en la configuración $I_2 = (L'_1, L'_2, L'_3)$ en cada paso de la computación. Por lo tanto, una computación es definida aquí como una secuencia de configuraciones I_0, I_1, \dots donde I_0 es la configuración

inicial descrita y la inducción o represión final del gen de fluorescencia en la bacteria AND marca el fin de la secuencia.

Cuadro 10.2: Reglas de inferencia de bajo nivel

R1	if <i>Arabinosa</i> then $pAraBAD_a = ACTIVE \wedge pAraBAD_b = ACTIVE$
R2	if <i>IPTG</i> then $pLac = ACTIVE \wedge \Delta pLac = INACTIVE$
R3	if $pAraBAD_a = ACTIVE \vee pLac = ACTIVE$ then $luxIa = TRUE$
R4	if $pAraBAD_b = ACTIVE$ then $algT = TRUE$
R5	if $\Delta pLac = INACTIVE$ then $mucAB = FALSE$
R6	if $algT = TRUE \wedge mucAB = FALSE$ then $\Delta pAlgT = INACTIVE$
R7	if $ AHL_b \geq \delta_c$ then $pLuxRb = ACTIVE$
R8	if $pLuxRb = ACTIVE$ then $luxRa = TRUE$
R9	if $luxRa = TRUE \wedge AHL_a \geq \delta_c$ then $pLuxRa = ACTIVE$
R10	if $pLuxRA = ACTIVE$ then $gfp = TRUE$
R11	if $luxI_i = TRUE$ then $ AHL_i _t > AHL_i _{t-1}$
R12	if $luxI_i = FALSE$ then $ AHL_i _t < AHL_i _{t-1}$

La secuencia de reglas de inferencia de bajo nivel para la situación inicial descrita anteriormente y que finaliza en expresión de fluorescencia es:

$$\{R1, R3, R4, R7, R8, R9, R10\} \quad (10.4)$$

Las dos últimas reglas de la tabla 11.2, R11 y R12, son reglas dinámicas que marcan el incremento y decremento, respectivamente, de moléculas AHL_i en la memoria compartida desde el instante anterior $t - 1$ al instante de tiempo actual t .

10.4. Simulación

Esta sección trata de corroborar el correcto funcionamiento del biodispositivo XOR mediante la simulación computacional del mismo, como fase esencial del proceso de validación al que todos los desarrollos están sometidos, y proporcionar conclusiones relevantes para la comprensión del comportamiento de la población. La

comunidad bacteriana multi-cepa a simular, estará compuesta por tres cepas lógicas genómicamente diferenciadas y diseñadas para colaborar e interactuar con las otras de forma que sólo al mezclar las tres cepas pueda emerger el comportamiento buscado.

El carácter emergente [Cla98] del biodispositivo celular surge de esta interacción distribuida entre cepas en la que el comportamiento final no puede entenderse desde el punto de vista de cada componente por separado. Obviamente, durante el estudio del diseño de la cepa OR es imposible adivinar el futuro comportamiento de la población XOR si se desconoce la naturaleza de las dos cepas restantes. Hoy en día, el concepto de emergencia es objeto de gran atención y estudio en ramas científicas como las ciencias de la computación (por ejemplo, los muchos desarrollos en *swarm intelligence* o inteligencia de enjambre [Dor99]⁵) o la filosofía de la biología.

El emergentismo se opone así a la visión mecanicista de la biología que defiende su carácter mecánico de autómatas. En el paradigma computacional de puertas celulares que se está presentando en este capítulo se podría entender desde este enfoque mecanicista el funcionamiento de una bacteria individual pero nunca el comportamiento social resultado de la interacción célula-célula.

Para la simulación se ha hecho uso de la aplicación Simulink⁶ de Matlab que permite modelar sistemas dinámicos y la posterior visualización de su comportamiento. La librería BioBrick [Kid07] ha servido como punto de partida e inspiración para el diseño, así como han sido necesaria la aplicación de varias ecuaciones en el dominio del tiempo⁷ que simularan los procesos moleculares.

⁵Han adquirido mucha relevancia durante los últimos años los algoritmos basados en el comportamiento de las hormigas (Ant Colony Optimization) o en el comportamiento de las abejas (Bee Optimizatio).

⁶<http://www.mathworks.com/products/simulink/>

⁷La página web <http://openwetware.org> ofrece gran variedad de información sobre el comportamiento de procesos moleculares que han servido de punto de partida para la simulación computacional. El proyecto *OpenWetWare* representa una herramienta útil entre los investigadores para el intercambio de información.

10.4.1. Construcción del modelo

Cada cepa se diseña como un bloque independiente con dos entradas ($In1$, $In2$) y una salida ($Out1$). Las conexiones entre estos bloques y su posterior lectura pueden observarse en la figura 10.3 que representa el modelo del diseño de alto nivel de la figura 10.2.1. El estado del circuito representado corresponde a la introducción en la población de una sola de las entradas, Arabinosa.

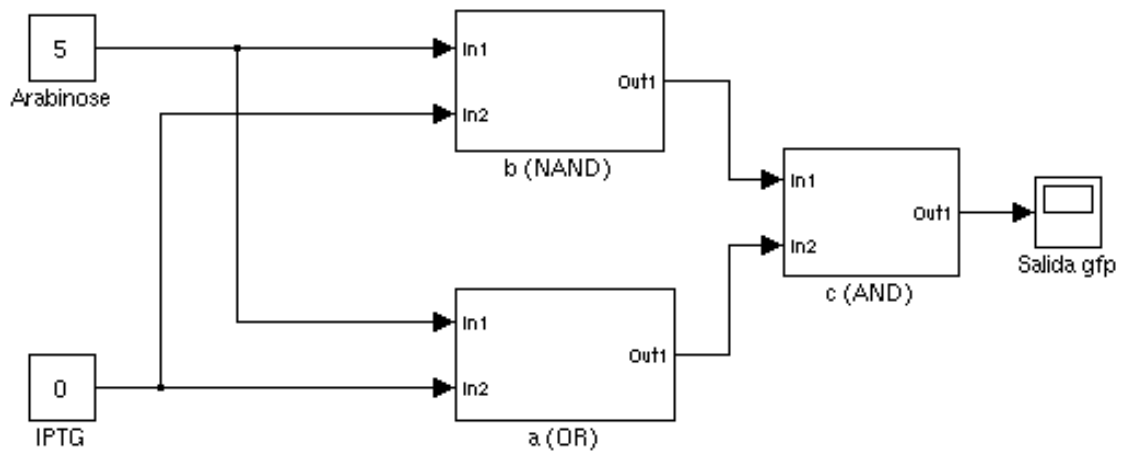


Figura 10.3: Diseño de alto nivel del circuito simulado

Se ha considerado como referencia en cuanto a la concentración molecular en la memoria compartida del biocircuito el conjunto numérico $\{0, \dots, 5\}$ como representación de la producción transcripcional total de una cepa bacteriana. De esta forma, una cepa que no está produciendo moléculas del tipo AHL_i se la asigna un valor nulo mientras una cepa con producción activa nunca supera el valor 5 que es el máximo nivel de concentración molecular que puede producir.

Esta producción variable en el tiempo es la que, junto con la determinación de los umbrales, marca la sincronización necesaria en la población. En este caso, se ha diseñado el umbral únicamente en las membranas de la cepa AND para reaccionar sincronamente a un determinado nivel de salida de las cepas NAND y OR.

Esa es la razón por la que en la figura 10.3 se introduce directamente una con-

centración máxima (nivel 5) de Arabinosa en la comunidad heterogénea forzándose así la reacción de los individuos de la capa de entrada en, aproximadamente, el mismo instante de tiempo.

El diseño del bloque correspondiente a la bacteria OR es detallado en la figura 10.4. La característica más relevante de este modelo es el tratamiento lógico de la función OR en el proceso de recorrido de la señal desde las salidas de los promotores al gen estructural *luxIa*, el cual confiere a la puerta celular un leve carácter analógico o dinámico frente a la naturaleza estrictamente digital de los circuitos con alimentación eléctrica. En lugar de hacer llegar al gen un valor digital de inducción o represión se calcula un valor analógico de carácter aditivo, de manera que tres valores distintos pueden llegar a él:

- Un valor nulo correspondiente a la situación en la que ninguna de las entradas se encuentra presente
- Un valor medio correspondiente a la presencia de una sola de las entradas. Se entiende aquí por valor medio, el valor normal de traducción que tiene un promotor aislado.
- Un valor alto que hace referencia a la presencia de ambas moléculas. Ese valor alto es consecuencia directa de la actividad de dos promotores sobre el mismo gen estructural.

El estudio de los rangos dinámicos de expresión mediante el control de una tupla de promotores consecutivos es un importante desarrollo para el diseño de complejos mecanismos transcripcionales. Una relevante característica de este estudio [Lam08] es la capacidad de la cromatina para inhabilitar la unión de factores de transcripción en el genoma con la consecuente regulación de la expresión genética.

Aunque en el diseño de la cepa OR se incluye un control básico de este estilo, no se utiliza éste, siendo la salida de dicha cepa leída de un modo binario y digital. Sin embargo, esta característica muestra, nuevamente, la escalabilidad del biocircuito.

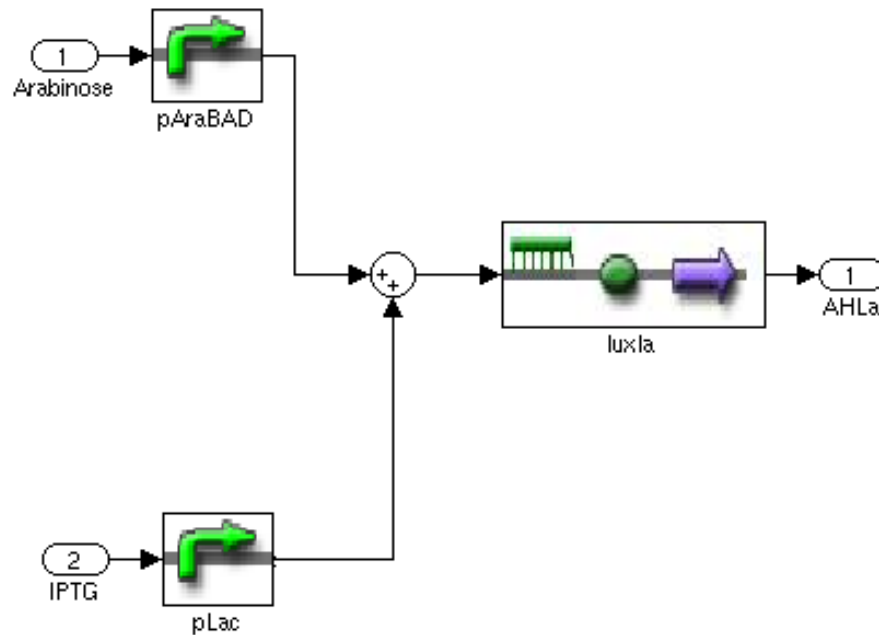


Figura 10.4: Diseño interno de cada bacteria OR

Cada bacteria perteneciente a la cepa NAND es modelada para el proceso de simulación tal y como se observa en la figura 10.5. La representación de las interacciones entre la expresión de los genes estructurales *algT* y *mucAB* es el aspecto más complicado del diseño de estas puertas celulares y merece especial atención. En la figura se distinguen tres complejos promotor-gen:

- En la parte superior de la figura 10.5 se encuentra el par de expresión formado por el promotor pAraBAD, que es un promotor inducible por Arabinosa, y el gen *algT* que es expresado por el promotor anterior. Los factores de transcripción resultantes actúan de factores represivos del promotor truncado $\Delta pAlgT$, a través del componente *Switch1* según se explica a continuación. Estos factores son desactivados (pierden su capacidad represora) por las moléculas resultantes de la expresión del gen *mucAB* lo que se representa mediante el bloque sumatorio. Así, cuando ambos factores de transcripción están presentes, la señal que llega a la entrada central del *Switch1* será prácticamente nula.
- En la parte media de la misma figura se representa el complejo de promoto-



⁸La empleo de la palabra difusa *prácticamente* hace referencia a la necesidad de establecer un umbral en la decisión del bloque *Switch1* debido al posible ruido genético generado en el interior de la bacteria.

actúa como señal inhibidora de la expresión genética.

- La cadena genómica modelada en la parte inferior simula el complejo de promotores $pNm\Delta pLac$ y el gen estructural *mucAB*. Al igual que la anterior, esta subfigura basa su funcionamiento en un switch, en este caso el bloque *Switch*. La decisión de éste depende de la presencia o ausencia del valor de entrada IPTG. Si no se ha introducido esta molécula en la población, el *Switch* seleccionará la entrada de la derecha. Esto provoca el funcionamiento normal del promotor constitutivo. En otro caso, el bloque seleccionará la entrada de la izquierda y provocará la consecuente represión del promotor truncado $\Delta pLac$ sobre el gen. Como se indicó previamente, la expresión de este gen resulta en la neutralización de cualquier factor de transcripción algT .

La variable constante situada a la entrada de cada promotor constitutivo pNm determina su carácter constante. Se ha considerado oportuno separar esta variable de la descripción interna del promotor para poder variarla fácilmente. Aquí, el valor que le da a la variable es de 5, que es el valor máximo de expresión considerado en esta simulación.

La figura 10.6 muestra el diseño interno de las bacterias pertenecientes a la cepa AND. También aquí se distinguen tres conjuntos promotor-gen que, junto a los dos switches, definen el comportamiento lógico AND característico de esta cepa:

- En la parte superior se colocan secuencialmente los bloques correspondientes a la constante ($Nm3$) que sirve como entrada de activación al promotor constitutivo, el propio promotor constitutivo ($pNm3$) y el gen estructural *luxRb*.
- En la parte central se sitúan el conjunto promotor-gen $pLuxRb-luxRa$ y el switch *SwitchAHLb*. El comportamiento de éste último es más intuitivo que switches de la figura 10.5 y refleja de una manera más directa el diseño inicial establecido en la sección 10.3 de este capítulo. Dicho bloque selecciona de una manera condicional cuál va a ser la entrada al promotor inducible $pLuxRb$ mediante el nivel de moléculas AHL_b de entrada. Así, al situarse dicho nivel por encima del umbral δ prefijado para todos los individuos de la cepa

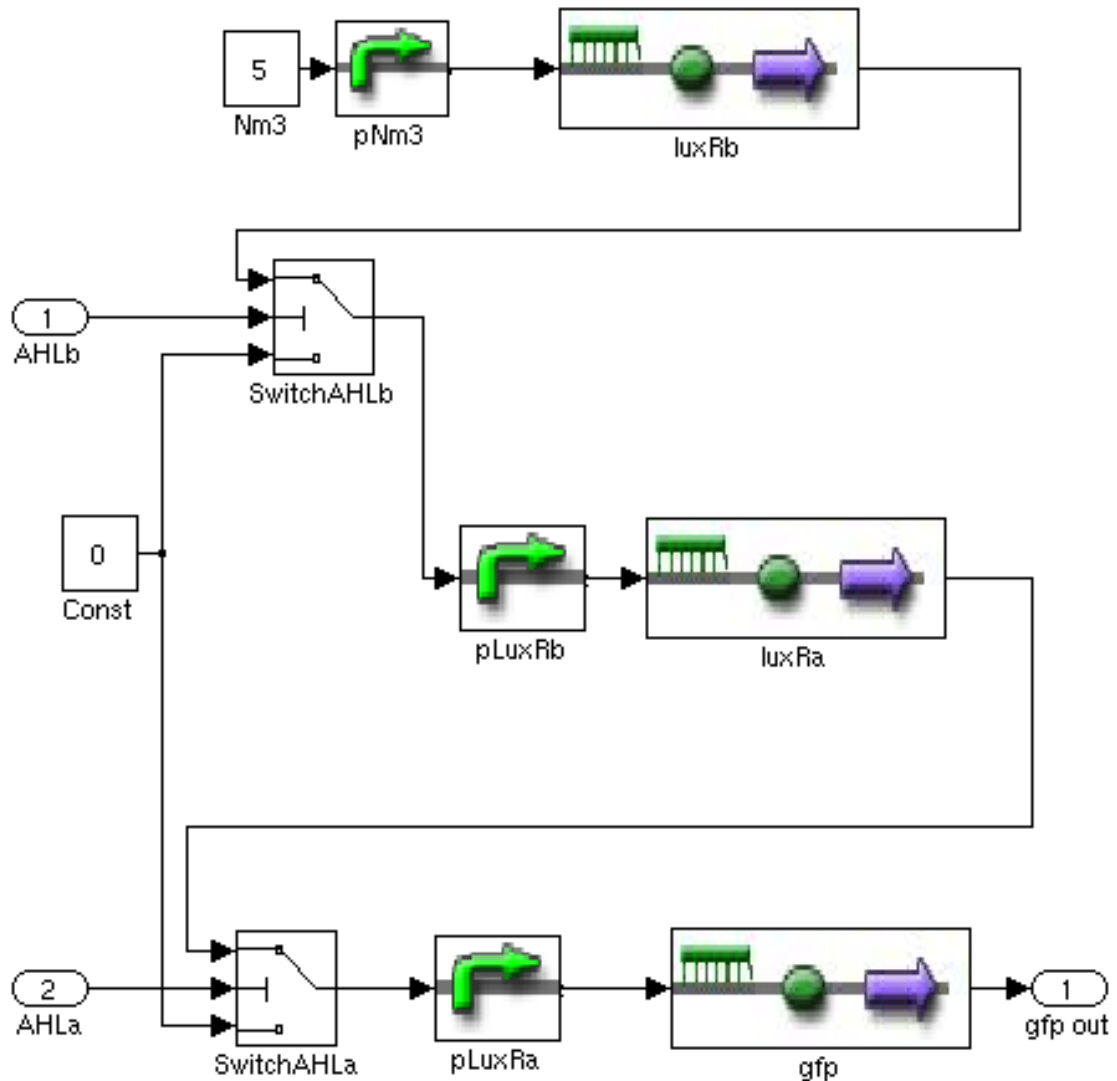


Figura 10.6: Diseño interno de cada bacteria AND

AND, el switch seleccionará la entrada superior activando así los factores de transcripción provenientes del gen *luxRb* y la consecuente inducción del promotor pLuxRb. Si el nivel de concentración de AHL_b en la memoria compartida no supera el umbral de condición, el switch seleccionará la entrada inferior simbolizando así la ausencia del proceso de activación de los factores de transcripción provenientes del gen *luxRb* y como resultado el promotor pLuxRb no se activará.

- De manera análoga, la parte inferior de la figura 10.6 muestra el control transcripcional del gen de fluorescencia (*gfp*) por parte del switch *SwitchAHL_a* dependiendo del nivel de moléculas AHL_a en la memoria compartida, lo que se traduce en la inducción o no del promotor inducible pLuxRa. La expresión final del gen de fluorescencia verde en una bacteria provoca la fluorescencia de la misma. Esta señal lumínica es considerada como una salida satisfactoria (o TRUE, en lógica booleana) de la puerta celular AND y, por ende, del circuito XOR en su totalidad.

10.4.2. Resultados

Ocho simulaciones son llevadas a cabo correspondiendo con los cuatro casos de entrada del circuito y con el fin de establecer conclusiones relevantes para realizar un estudio del comportamiento de la comunidad XOR. Las simulaciones se pueden agrupar en dos grupos:

- Simulaciones de salida final: en las que se pone de relevancia la salida última del circuito.
- Simulaciones de computación completa: en las que se muestra la salida de cada una de las cepas lógicas con propósito de estudiar la finalidad de cada comportamiento individual.

En la tabla de verdad 11.3 se muestran las salidas teóricas en modo booleano de un circuito XOR. Se puede comprobar como la salida esperada coincide con la salida simulada de las figuras 10.7 y 10.8.

Esta última figura, en la que se incluye la salida de las tres cepas, merece especial atención. En los casos de entrada 00, 01 y 10 el comportamiento de las tres cepas es el esperado. Sin embargo, llama la atención el comportamiento de las bacterias NAND en la simulación correspondiente al caso de entrada 11 (figura 10.8d). Este resultado es debido a la programación paralela con la que se ha diseñado esta puerta celular.

Cuadro 10.3: Tabla de verdad del biocircuito XOR

Arabinosa	IPTG	GFP	Sim. Final	Sim. Completa
FALSE	FALSE	FALSE	Figura 10.7a	Figura 10.8a
FALSE	TRUE	TRUE	Figura 10.7b	Figura 10.8b
TRUE	FALSE	TRUE	Figura 10.7c	Figura 10.8c
TRUE	TRUE	FALSE	Figura 10.7d	Figura 10.8d

Para este caso, el valor que debe tomar la bacteria NAND ha de ser *FALSE*, es decir, la célula no debe producir moléculas AHL. Al contrario, al principio de la computación tal y como se muestra en la figura 10.8d el nivel de moléculas expresado por esta cepa se incrementa. Si recordamos el diseño de dicha bacteria, es un promotor constitutivo pNm el encargado de regular la transcripción del gen estructural *luxIb* y son las entradas las encargadas de inhibir o no dicho promotor. La campana que describe el comportamiento de las bacterias NAND es, por tanto, resultado del tiempo transcurrido desde el inicio de la computación⁹ hasta el procesamiento de las entradas (en este caso la expresión del gen *algT*), momento en el cual la producción de moléculas AHL se detiene.

Este intervalo de tiempo indica la necesidad del correcto establecimiento de los umbrales de sincronización en la cepa AND. En este caso, el umbral debe ser siempre un valor superior al valor máximo que alcance la campana. Es decir, considerando la expresión de moléculas AHL por parte del gen *luxIb* previas al proceso de las entradas como ruido de interferencia del biocircuito, el umbral debe servir como medio de paliación de este ruido.

Así como el ruido genético ha sido muy estudiado en el interior de una célula independiente [Hoo06], la construcción y diseño de poblaciones heterogéneas que persigan un fin conjunto obliga a tratar el ruido provocado por concentraciones de datos inoportunas en la memoria compartida del dispositivo. La definición de umbrales de sincronización es, como conclusión, un buen punto de partida para esa tarea.

⁹Entendiendo como tal la introducción de moléculas de entrada.

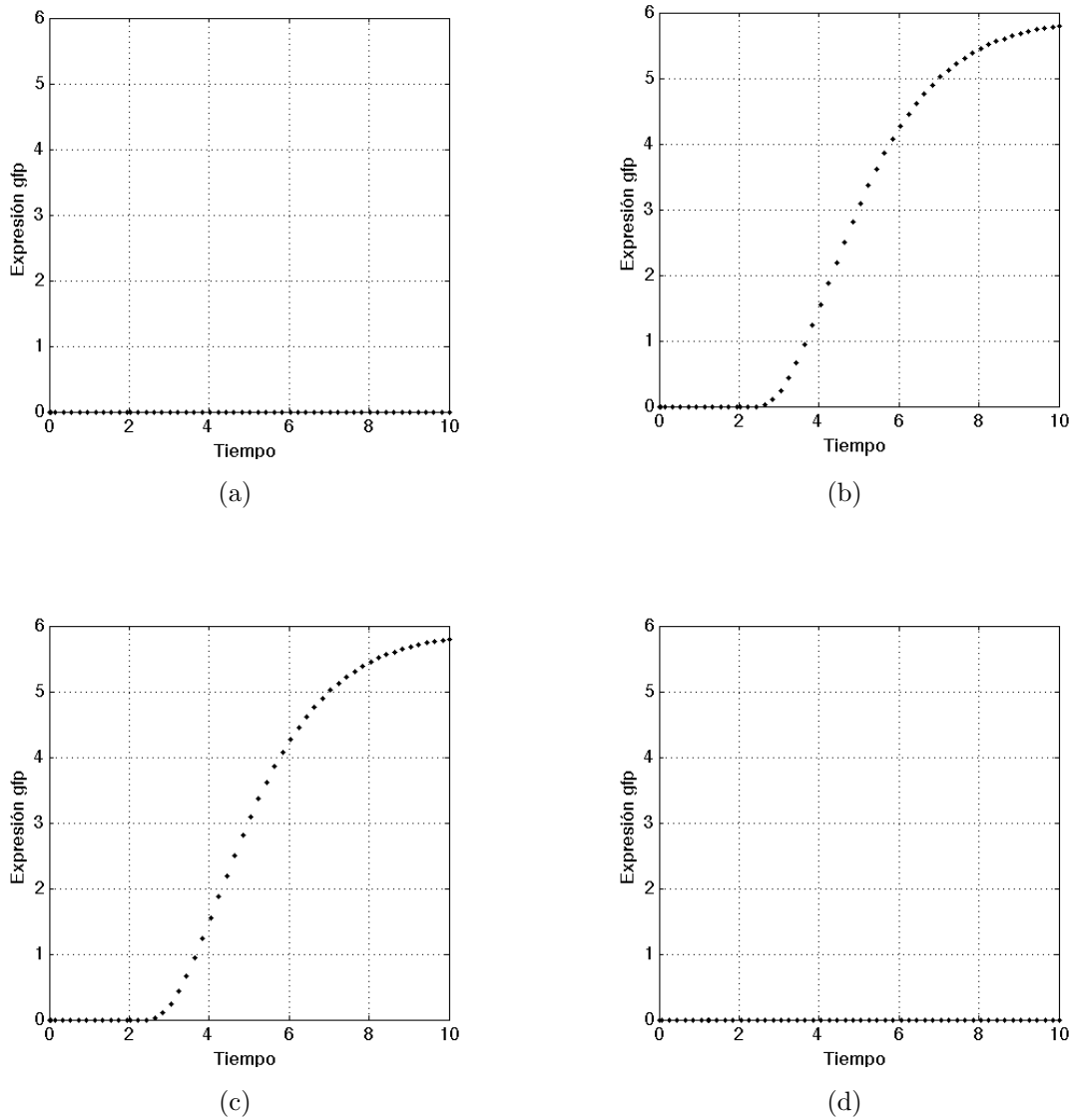


Figura 10.7: Gráficos de verdad correspondientes a la salida final del circuito. a) cómputo del caso 00; b) cómputo del caso 01; c) cómputo del caso 01; d) cómputo del caso 11.

Otra importante conclusión resulta del paradigma de programación paralelo que guía el comportamiento de la cepa NAND. En efecto, al encontrar más de un promotor constitutivo pNm en el interior de cada bacteria de dicha cepa podemos afirmar que existen más de una instrucción computacional en su diseño. En concreto, las instrucciones por bacteria son:

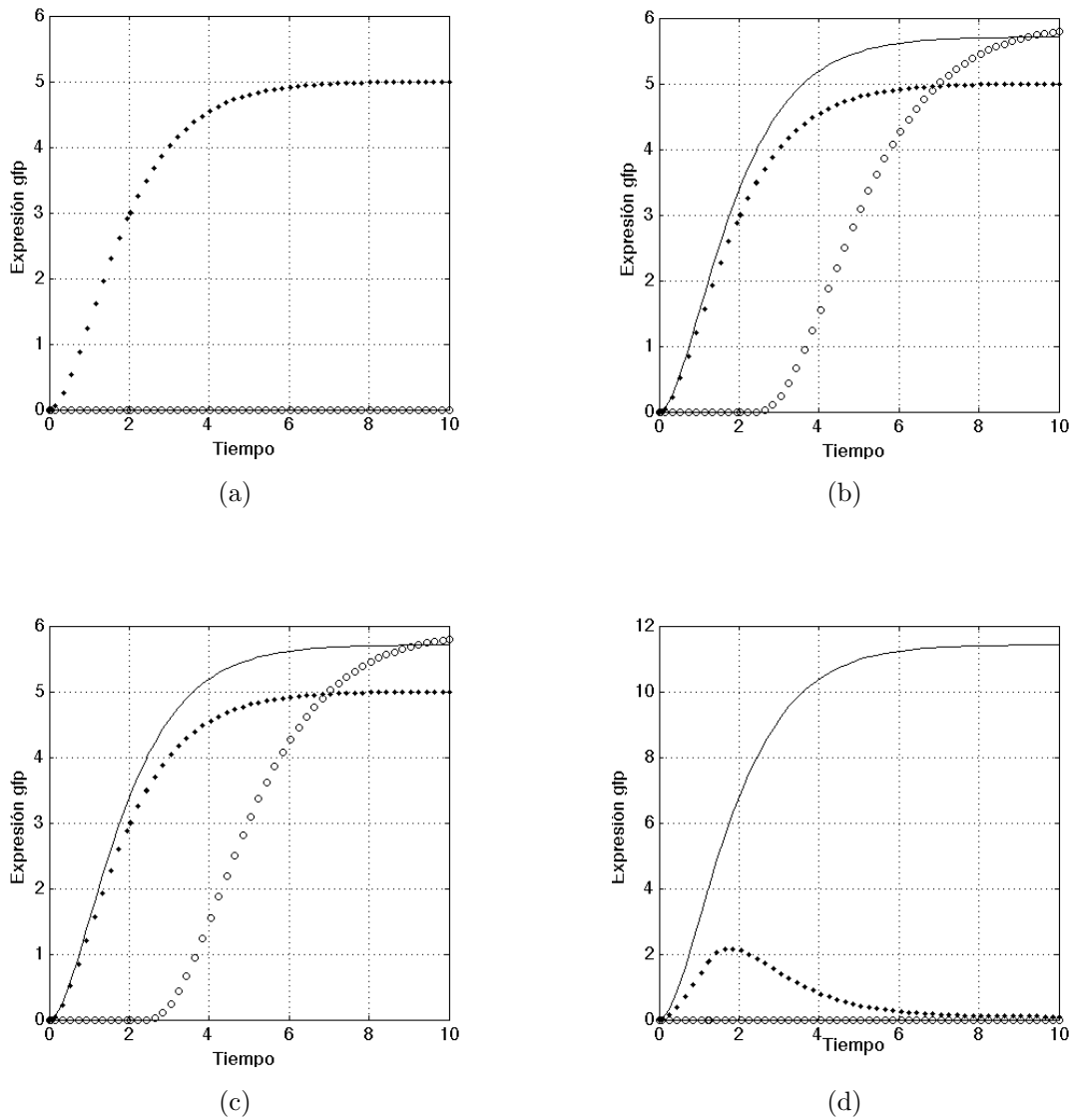


Figura 10.8: Expresión completa de las tres cepas: salida de la cepa NAND (●); salida de la cepa OR (—); salida de la cepa AND (○). a) cálculo del caso 00; b) cálculo del caso 01; c) cálculo del caso 01; d) cálculo del caso 11.

$$|Instrucciones| = \begin{cases} \sum_{i=1}^n i & \text{si } n \neq 0 \\ 1 & \text{en otro caso} \end{cases} \quad \text{siendo } n \text{ el número de pNm}$$

Ya que cada instrucción comienza a procesarse por separado y al inicio de la computación, podemos tener varias instrucciones ejecutándose en paralelo. Esta

característica, que únicamente se encuentra en la bacteria NAND, es también un indicador de la escalabilidad del sistema en lo que al caracter paralelo y secuencial se refiere.

10.4.3. Discusion

Futuros trabajos que traten de aplicar este modelo deben encontrar el sustrato óptimo para albergar una comunidad heterogénea de bacterias como la descrita. Aparte de posibles aplicaciones del biodispositivo en organismos vivos, el proceso de diseño y testeo debe pasar obligatoriamente por una superficie artificial en laboratorio.

Ya se han desarrollado varios trabajos describiendo diferentes sustratos orgánicos (hidrogel) e inorgánicos (sol-gel) para estabilizar las bacterias [Bje06] así como métodos de láser para situar bacterias en formaciones específicas tales como matrices en distintas superficies [Bar04]. Alterando la estructura del medio, la composición del sustrato y la fisionomía del circuito pueden conseguirse modificaciones sustanciales que deriven en características muy ventajosas del biodispositivo.

Respecto a la fisionomía del circuito, grandes avances en el coste computacional de los algoritmos implementados con bacterias pueden conseguirse con las técnicas basadas en láser. Esto es debido a la capacidad que otorga este método de situar las bacterias en sitios fijos y determinados del espacio sobre una superficie rígida. Así, se posibilita la configuración de estructuras complejas de redes de bacterias limitando sus el número de vecinas de éstas según los requisitos del enunciado.

Sin embargo, la desventaja estaría relacionada con el carácter *in-vivo* de las aplicaciones que se deseen diseñar ya que este método limita el tipo de superficie que debe sostener la población bacteriana. El biocircuito propuesto en este capítulo no presenta restricciones respecto a superficie que sirva de sostén a la comunidad, aunque el número de vecinos de cada bacteria sea variable e imprevisible durante la computación.

La unión de las dos cualidades, el control de la fisionomía de la población y

la versatilidad de ésta en distintos entornos, es un objetivo aún inalcanzado pero prometedor.

Capítulo 11

Diseño de un oscilador genético

Índice

11.1. Arquitectura del Modelo	133
11.2. Diseño de alto nivel	138
11.3. Diseño de bajo nivel	143
11.4. Simulación	148

En este capítulo se presenta y propone un diseño ideado para poblaciones de bacterias donde múltiples cepas bacterianas interactúan y colaboran con el fin de alcanzar un comportamiento oscilatorio repetido basado en transiciones sincrónicas entre los colores rojo y verde.

11.1. Arquitectura del Modelo

Como ya se ha visto en capítulos anteriores, varios modelos de osciladores genéticos han sido probados con éxito en experimentos in-vivo utilizando las células aisladas como dispositivos de procesamiento en los que la computación (el comportamiento oscilatorio en este caso) se completa y termina. Es decir, estos experimentos obtienen múltiples ejemplos de la computación deseada, tanto como bacterias contenga la población o cepa en que se esté trabajando. Siendo una cepa C definida de

la forma:

$$C = \{B_1, \dots, B_n\} \quad (11.1)$$

donde B representa cada célula y n el número de células en la población, entonces cada B_i representará un oscilador independiente.

Sin embargo, el diseño aquí propuesto se adecua al objetivo de búsqueda de la coordinación necesaria entre varias cepas bacterianas que conforman una misma población heterogénea P :

$$P = \{C_1, \dots, C_m\} \quad (11.2)$$

donde C es cada una de las cepas del biodispositivo P y m el número de éstas que lo forman.

Esta coordinación entre cepas, fundamentada en la modificación de las comunicaciones célula-célula, tiene como finalidad conseguir una funcionalidad específica y emergente de la población P que no es posible conseguir con un subconjunto de dichas cepas. El comportamiento oscilatorio será, por tanto, único para toda la población encontrándose así una única computación completa dentro de la comunidad heterogénea de células. La población diseñada cuya explicación aborda este capítulo está formada por un conjunto de $m = 3$ cepas modificadas genéticamente de forma diferente según se expondrá más adelante.

Al igual que en el desarrollo del biocircuito XOR en el capítulo anterior, el control sobre la comunicación célula-célula es el aspecto característico y fundamental que permite el diseño del sistema. De este modo se alteran y modifican las rutas químicas de comunicación basadas en el intercambio de moléculas de la familia N-Acylhomoserine lactone (AHL) y los umbrales que permiten la reacción de las células receptoras con el fin de diseñar cepas con funcionalidades complementarias.

El comportamiento esperado de dicha comunidad heterogénea formado por cepas modificadas transgénicamente representa un novedoso paradigma de regulación de los ciclos celulares y un importante desarrollo en el ámbito de las comunicaciones entre células sintéticas. El hecho de poder alterar las características de una bacteria hasta el punto de crear células con genomas mínimos en los que se incluye informa-

ción genética específica que conlleva la formación de células con comportamientos muy diferentes, hace pensar en un nuevo concepto: especies artificiales.

En efecto, para el diseño de un dispositivo multi-cepa no es necesario el uso de distintas especies bacterianas ya que se puede recrear la misma diferenciación en cepas sintéticas.

Al ser el funcionamiento autónomo uno de los objetivos más importantes a alcanzar por este desarrollo, el diseño de la comunidad bacteriana debe incluir mecanismos de auto-regulación y sincronización que puedan evitar la intervención humana lo máximo posible.

Una vez la computación haya comenzado, la cooperación y coordinación entre las distintas cepas bacterianas debe llevarse a cabo *in-vivo* por las propias cepas de manera que las tareas que desempeñen sean totalmente independiente de factores externos a la comunidad multi-cepa.

La autonomía en el diseño de los biodispositivos sintéticos es una característica altamente deseada ya que, en un futuro, la utilización de éstos estará relacionada con ramas científicas distintas como pueden ser la ecología o la medicina en las que la intervención humana es claramente restringida. Tal sería el caso de la aplicación de estos modelos a medicamentos o biosensores ecológicos.

En los últimos años se está acrecentando la investigación acerca de sensores biológicos basados en bacterias. La capacidad que tienen éstas de medir y sentir el entorno resulta muy provechosa para realizar aplicaciones en las que se las fuerce a medir la bioviabilidad y toxicidad de un medio específico [Bje06]. Sin embargo, el desarrollo de células transgénicamente modificadas para cumplir funciones sensitivas se ha llevado a cabo tradicionalmente desde el punto de vista de la robustez en el diseño. Así se da prioridad a conseguir bacterias con funcionalidades sencillas pero que formen biodispositivos estables en los entornos que se pretenda monitorizar¹.

¹El proceso de monitorización hace referencia a la lectura y supervisión del estado del entorno para sincronizar la ejecución de la bacteria en relación a un estado concreto y encaminada al logro de un fin específico.

También se invierte mucho esfuerzo en la construcción de superficies físicas en las que situar las bacterias con funciones biosensitivas como biochips o fibras ópticas especializadas que mejoren las prestaciones del biodispositivo. Todos estos desarrollos mantienen un prometedor potencial en el futuro a la espera, entre otras cosas, de un mayor conocimiento de los microorganismos a manipular y unos diseños celulares novedosos.

Los biosensores basados en bacterias se fundamentan en las respuestas de éstas a estímulos externos en condiciones ambientales específicas y pueden englobarse en dos grandes tipos dependiendo de la filosofía de diseño:

- Sensores de transición *on* \rightarrow *off*. Las células bacterianas actúan como promotores constitutivos produciendo una señal constante y medible (*reporter*). La toxicidad del medio, en este caso, es estimada según el nivel de inhibición de dicha señal.
- Sensores de transición *off* \rightarrow *on*. En su estado normal, las bacterias inducibles usadas en estos sensores, no emiten ningún tipo de señalización. En este caso la toxicidad del medio o la situación de estrés que se desea medir, hace activar los promotores inducibles de las bacterias.

En ambos casos, los genes usados como señalizaciones medibles para poder distinguir los niveles de estrés que el biosensor está midiendo pueden ser varios. Comunmente se usan: *lacZ*², *luc*³, *luxAB*⁴ y *gfp*⁵, siendo esta última -como se ha visto- la salida más usada en los desarrollos propuestos en este trabajo.

Desde el punto de vista de la computación, las bacterias de estos biosensores forman procesadores relativamente sencillos tanto en el número de instrucciones como en la arquitectura de comunicaciones global de la comunidad. Es por esto, que los desarrollos producidos en los últimos años por la Biología Sintética pueden

²Codifica β -galactosidase.

³Codifica luciferasa eucariota.

⁴Codifica luciferasa bacteriana.

⁵Codifica la proteína de fluorescencia verde.

encontrar en los biosensores un acercamiento a problemas ecológicos distanciándose así de las aplicaciones matemáticas que sirven de trasfondo a muchos de estos diseños.

Dejando aparte los objetivos ecológicos y centrándonos nuevamente en el modelado celular con inspiración computacional, se propone a continuación una arquitectura de comunicaciones entre bacterias que lleva a la población a desempeñar una función básica y fundamental en los computadores tradicionales: la señal síncrona de reloj.

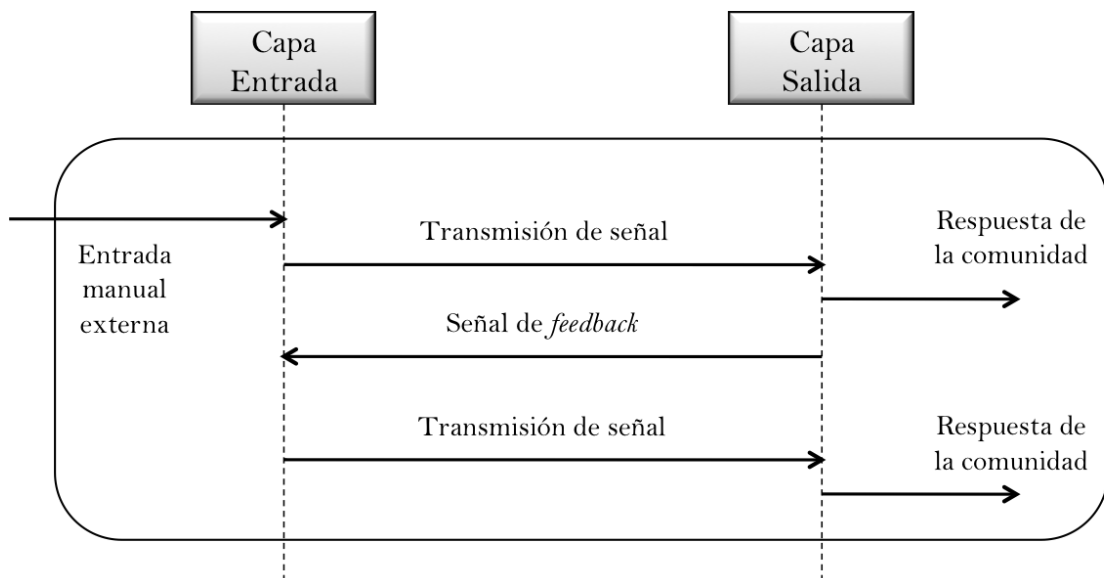


Figura 11.1: Arquitectura de comunicaciones diseñada en la población de bacterias multi-cepa con comportamiento oscilatorio.

Se han visto ejemplos en capítulos anteriores de comunidades heterogéneas de bacterias transgénicas diseñadas para responder al funcionamiento de arquitecturas de comunicaciones simples en las que la comunicación entre cepas fluye únicamente en dirección *capadeentrada* \rightarrow *capadesalida* y la capa de entrada debe ser estimulada desde el exterior cada vez que se desea obtener una respuesta.

En contraposición a éstos modelos, la comunidad descrita en este capítulo sigue la arquitectura mostrada en la figura 11.1 donde se observa cómo el comportamiento autónomo se consigue mediante el uso de señales de *feedback* entre las capas

de entrada y salida, lo que lleva a la comunidad a mostrar múltiples respuestas sin la necesidad de intervención externa (excepto la inicial). Esta división por capas establece roles bien definidos entre las cepas de la comunidad, distinguiéndose funcionalidades similares entre las cepas de la misma capa. La catalogación en capas de las bacterias es una abstracción de inspiración computacional que resulta muy útil para formalizar las arquitecturas a diseñar.

Para el diseño del oscilador, esta segunda arquitectura de comunicaciones usa una cepa para formar la capa de entrada encargada de sincronizar las señales de *feedback* y dos cepas con funcionalidades distintas para formar la capa de salida. Aún siendo ésta una arquitectura más compleja, sigue siendo relativamente básica especialmente si la comparamos con las arquitecturas desarrolladas para intercambio de datos entre distintas máquinas o procesadores electrónicos.

Siguiendo esta línea de diseño, futuros desarrollos podrán expandir este modelo mediante la introducción de capas intermedias⁶ y el aumento o disminución de cepas celulares en la composición de las distintas capas con el propósito de crear arquitecturas más complejas.

11.2. Diseño de alto nivel

En analogía con los computadores digitales convencionales, este diseño está basado en modelo de comunicación cliente/servidor en el que las bacterias de la capa de entrada actúan como servidoras y las bacterias de la capa de salida como clientes. Siguiendo esta analogía también podemos identificar en la solución nutritiva que sirve de soporte a la comunidad de bacterias, al agente intermedio que desempeña el papel de *buffer* o memoria compartida y que facilita la transmisión de señales entre los distintos procesadores del modelo. Esta especial memoria común es general para todas las bacterias en el sentido en que todas ellas realizan procesos de *lectura*

⁶Las Redes de Neuronas Artificiales también basan su morfología en la distribución de neuronas en varias capas dependiendo del modelo implementado y pueden servir como ejemplo en el diseño de arquitecturas de comunicaciones célula-célula más complejas.

sobre la memoria compartida de manera constante utilizando su membrana como dispositivo lector y *búscan* aquellos datos para los que han de reaccionar siguiendo su programación genética.

Como paso posterior a esa lectura de datos, cada célula *escribe* en este *buffer* la información de salida producida como respuesta. Ya que cada bacteria actúa como un procesador independiente en este modelo, existe también una memoria privada para cada una. Esta memoria local corresponde al espacio delimitado por la membrana en el que las funciones y operaciones locales se llevan a cabo.

La comunicación entre las distintas cepas se basa, por tanto, en procesos concurrentes de lectura/escritura sobre la solución nutriente. En este espacio intercelular compartido se encuentra la información relativa a la comunicación y sincronización entre las bacterias en forma de moléculas de la familia AHL.

Para realizar un análisis de la computación a alto nivel lo único que interesa es conocer la naturaleza y concentración de las moléculas AHL que en un momento determinado ocupan la memoria común, para lo cual cada bacteria se considera como una *caja negra*⁷ sin ser importante su estructura interna.

El objetivo final, que es la obtención de una oscilación lumínica recurrente *verde-rojo*, puede ser simplificado a conseguir una secuencia concreta y variable en el tiempo de tuplas $L_i = [Bool, M]$ representando la ausencia o presencia de moléculas AHL y las concentraciones de dichas moléculas en el *buffer* compartido. Esta secuencia viene detonada por I_i como se muestra en la ecuación 11.3.

$$I_i = \langle [Bool, M]_{AHL_1}, [Bool, M]_{AHL_2}, \dots, [Bool, M]_{AHL_n} \rangle \quad (11.3)$$

que simbolizan la presencia o no (*Bool*) y la concentración molecular (*M*) de determinada molécula AHL. Habrá tantos pares en cada estado I_i como moléculas AHL sean necesarias para la sincronización del biodispositivo. Es para alcanzar esta computación que las bacterias son diseñadas para alterar el contenido de la memoria común en uno u otro sentido, basandose el cambio $I_i \vdash I_{i+1}$ entre un estado y el siguiente en la manipulación de dichos conjuntos.

⁷En referencia a los test de caja negra y blanca del ciclo de vida del software.

Formalmente se dice que la configuración $I_1 = (L_1, \dots, L_2)$ cambia en la configuración $I_2 = (L'_1, \dots, L'_2)$ en cada paso de la computación. Por lo tanto, una computación es definida aquí como una secuencia de configuraciones I_0, I_1, \dots donde I_0 es la configuración inicial. Debido al caracter oscilatorio del biodispositivo, la secuencia de configuraciones no tiene límite teórico, aunque sí práctico como se verá en secciones posteriores.

El modelo cliente/servidor aquí diseñado representa una interacción entre células siguiendo un paradigma sencillo de paso de mensajes *rendezvous*⁸ en el que los clientes siempre están listos para desempeñar su tarea y el servidor se encarga de intercambiar turnos entre ellos, de manera que una bacteria al finalizar su acción espera hasta recibir el turno de nuevo.

Son cuatro los tipos de moléculas de la familia AHL que se van a usar en el diseño de este oscilador:

- AHL_g : Producida por la cepa servidora. Esta molécula resulta de utilidad para la cepa cliente capaz de expresar luz verde (*green*, en inglés) que la identifica como el permiso necesario para iniciar turno.
- AHL_r : Producida por la cepa servidora. Esta molécula resulta de utilidad para la cepa cliente capaz de expresar luz roja (*red*, en inglés) que la identifica como el permiso necesario para iniciar turno.
- AHL_s : Producida por la cepa cliente capaz de expresar luz verde. Esta molécula resulta de utilidad para la cepa cliente que la utiliza como testigo del fin de turno.
- $AHL_{s'}$: Producida por la cepa cliente capaz de expresar luz roja. Esta molécula resulta de utilidad para la cepa cliente que la utiliza como testigo del fin de turno.

⁸La Interfaz de Paso de Mensajes (conocido ampliamente como MPI, siglas en inglés de Message Passing Interface) es un protocolo estandarizado de comunicación entre computadoras que ejecutan un mismo programa en un sistema de memoria distribuida.

Cuadro 11.1: Reglas de inferencia de alto nivel

R1	if $ AHL_s \geq \delta_S$ then $S \longrightarrow AHL_r$
R2	if $ AHL_{s'} \geq \delta_S$ then $S \longrightarrow AHL_g$
R3	if $ AHL_g \geq \delta_{C_g}$ then $C_g \longrightarrow AHL_s \wedge gfp$
R4	if $ AHL_r \geq \delta_{C_r}$ then $C_r \longrightarrow AHL_{s'} \wedge rfp$

Como ya se ha mencionado, son tres las cepas que conforman los componentes de este biodispositivo y cuyo comportamiento se describe a continuación:

- *Cepa Servidora.* A cada célula perteneciente a esta cepa las llamaremos a partir de ahora bacteria central o servidora (S). Esta cepa sería el único componente de la capa de entrada de la arquitectura propuesta y tiene como función el producir señales alternas con el fin de estimular el funcionamiento de las cepas clientes de forma síncrona. Cuando la cepa servidora produce las moléculas necesarias para activar la acción de una de las cepas clientes, espera hasta la recepción del testigo y concede entonces el turno de expresión a la otra cepa cliente. Por lo tanto, cada bacteria central lee el *buffer* en busca de AHL_s o $AHL_{s'}$ y se encarga de producir AHL_g o AHL_r .
- *Cepa Cliente Verde y Cepa Cliente Roja.* Estas dos cepas forman la capa de salida, una para cada color de la oscilación. Las bacterias con capacidad de fluorescencia verde y roja se llamarán C_g y C_r respectivamente y se encargan de leer el *buffer* constantemente esperando el comienzo de su turno. Cuando detectan dicho comienzo, lo que quiere decir que cada bacteria central está produciendo moléculas AHL (AHL_g para turno de C_g o AHL_r para turno de C_r), lanzan la expresión genética de *gfp* o *rfp* dependiendo de la cepa y notifican el fin de su turno a la cepa servidora de la capa de entrada mediante la producción de AHL_s o $AHL_{s'}$ respectivamente.

Resulta de gran importancia la utilización de los umbrales (δ) en los procesos de lectura de cada membrana ya que las bacterias no tendrán que notar únicamente

la presencia o ausencia de una determinada molécula de comunicación (programación digital) sino también la concentración de éstas (programación analógica) para determinar el momento preciso de realizar su función. Si la salida producida por una determinada célula B_i debe usarse como entrada de otra B_j y el número de moléculas se representa por $|x|$, podríamos afirmar que cuando se cumple la ecuación:

$$|B_{i.out}| \geq \delta_{B_j} \quad (11.4)$$

la función de la bacteria B_j resulta activada por el vencimiento del umbral de B_i . El diseño de los correctos umbrales es fundamental para una buena gestión del ruido genético como se verá en la sección de simulación.

El modelado de estos umbrales permite la representación del funcionamiento del sistema a alto nivel con las cuatro reglas de inferencia de la tabla 11.1 en las que las premisas hacen referencia al incremento en la concentración de determinadas moléculas hasta superar el umbral de una cepa determinada, y las conclusiones son la producción de otra molécula por parte de dicha cepa. Cuando una regla no está activa la bacteria afectada detiene la producción de la molécula específica correspondiente, lo cual asegura la desaparición en la memoria compartida de esa información en un periodo de tiempo concreto y lineal.

Para establecer la secuencia de reglas de la tabla 11.1 que marca la ejecución de la computación (modificando, por tanto, la configuración de la ecuación 11.3) suponemos que el proceso comienza mediante la introducción manual en la comunidad del número necesario de moléculas $AHLs'$ que dispara la acción de la regla $R2$. Consecuentemente, la secuencia:

$$\langle R2, R3(gfp), R1, R4(rfp) \rangle \quad (11.5)$$

se repite indefinidamente, al menos en el plano lógico-teórico, mostrando así el comportamiento oscilatorio entre gfp y rfp ⁹ que se observa en la figura 11.2.

⁹Red Fluorescent Protein.

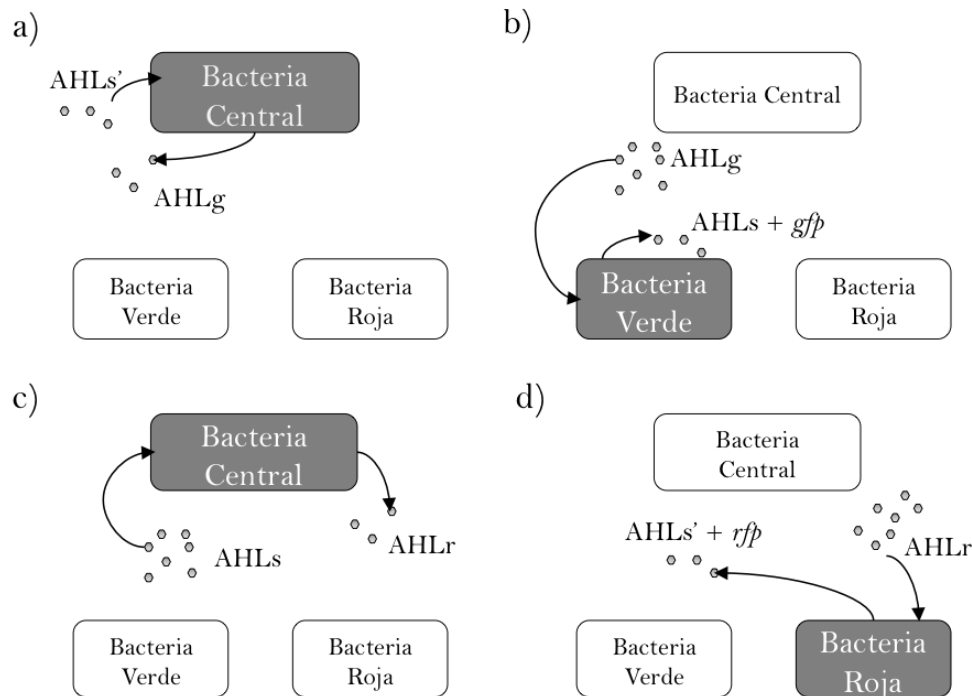


Figura 11.2: Secuencia de acciones que dan lugar a la oscilación lumínica. a) Situación inicial en la que se introduce $AHLs'$, estimulando a la bacteria central que concede turno a la bacteria verde; b) Turno de la bacteria verde que lleva a cabo la totalidad de su tarea y devuelve turno; c) La bacteria servidora o central recibe la señal de fin de turno y, como esta proviene de la bacteria verde, otorga el turno de proceso a la bacteria roja; d) Turno de proceso de la bacteria roja en la que emite la señal de fin de turno de ésta.

11.3. Diseño de bajo nivel

En esta sección se aborda el diseño interno de cada una de las tres cepas que componen el biodispositivo oscilatorio. La estructura propuesta está basada en la construcción de cascadas de señalización que deriven en la represión o inducción de determinados promotores para así poder hacer corresponder cada diseño a las entradas y salidas detalladas en la sección anterior.

Bases de datos como RegulonDB¹⁰ [Hue98], DBD¹¹ [Wil08] o PRODORIC¹²

¹⁰<http://regulondb.ccg.unam.mx/>

¹¹<http://dbd.mrc-lmb.cam.ac.uk/DBD/index.cgi?Home>

¹²<http://prodoric.tu-bs.de/>

[Mun03] son útiles herramientas por su dedicación a almacenar datos que aparecen en diferentes proyectos en genómica. Por lo tanto, hoy en día existen muchas posibilidades de configuración para construir unidades funcionales que sirvan como nuevos receptores o señalizaciones.

No sólo se hace uso de promotores inducibles o represibles para regular la expresión de los genes estructurales consecuentes. En este diseño es importante hacer hincapié en una de sus características más relevantes: la utilización de promotores híbridos.

Los promotores híbridos son promotores sintéticos capaces de ser inducidos por una molécula y reprimidos por otra diferente, como es el caso de los promotores *pTac* [Boe83] y *pSalCI434*¹³.

El primero se diseñó en los años 80 combinando secuencias del promotor *trp* y el *lac UV5*¹⁴, resultando un promotor híbrido con un comportamiento 5 veces más potente que el del promotor *trp* y 10 veces más potente que el de *lac UV5* lo que le convierte en un promotor muy útil para el control de la expresión genética. Éste puede ser reprimido por el represor *lac* o inducido por el inductor *isopropyl beta-D-thiogalactoside (IPTG)*.

El segundo promotor híbrido es de reciente síntesis (2009) y puede ser inducido por *salicylate* (sal del ácido salicílico) y reprimido por *CI434*¹⁵. Las ventajas de la utilización de estos promotores híbridos se mostrarán en la sección de simulación en la que se realiza una comparativa entre los resultados obtenidos del diseño sin utilizar dichos promotores y el diseño haciendo uso de ellos.

La funcionalidad particular de la bacteria central o servidora se detalla en la figura 11.3 donde toda la arquitectura interna de la memoria local de ésta se puede

¹³Este promotor ha sido diseñado por el equipo de Japón en el concurso internacional IGEM09. Toda la información se puede encontrar en la página web http://2009.igem.org/Team:PKU_Beijing/Team

¹⁴Diseñado usando la especie bacteriana *Escherichia coli* como soporte. Combina la región -35 del promotor *trp* y la -10 del promotor *lac*.

¹⁵De forma similar, se sustituye la secuencia entre la región -35 y la -10 del promotor inducible de *salicylate* por *CI434*.

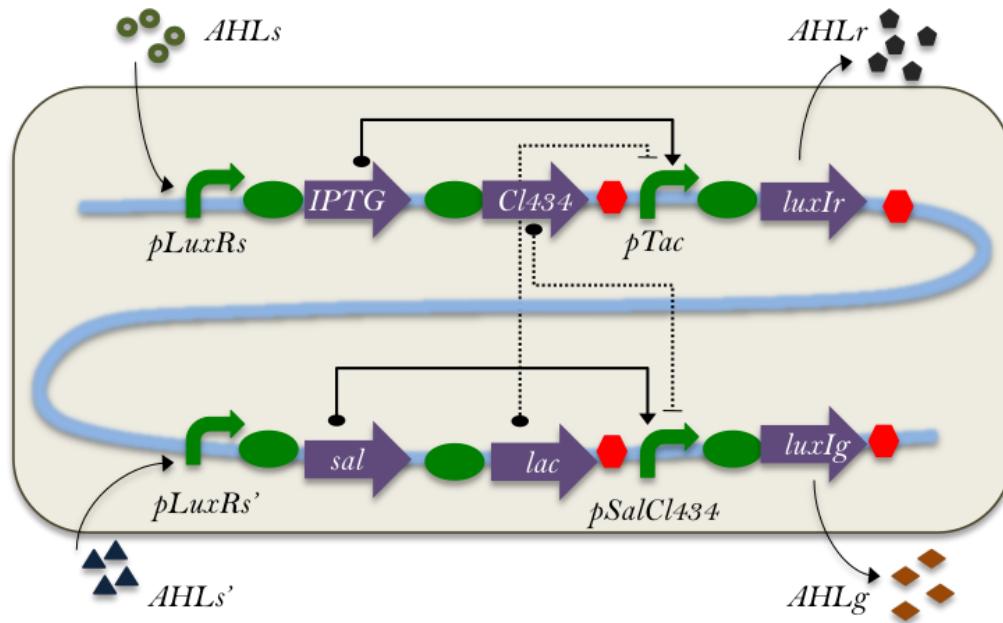


Figura 11.3: Diseño interno de la bacteria central.

observar gráficamente. Cuando la bacteria central *lee* a través de su membrana que la concentración de moléculas de AHL_s iguala o supera el nivel prefijado del umbral (δ), el promotor inducible $pLuxRs$ se activa dando como resultado la expresión de los dos genes estructurales consecuentes. La expresión del gen $IPTG$ se usa para activar la acción del promotor híbrido $pTac$ el cual se encarga de permitir, por parte del gen estructural $luxIr$, la producción de moléculas AHL_r , mientras que la expresión del segundo gen, $Cl434$ tiene como función la represión del segundo promotor híbrido $pSalCl434$ que inhabilita así la producción de AHL_g al no ser expresado el gen estructural $luxIg$. Con todo ello, la concentración de moléculas AHL_g que se encuentra en la solución nutritiva comenzará a decrecer a la vez que la concentración de moléculas AHL_r comenzará a aumentar.

Como resultado de esta cascada de señalización que se produce en el interior de las bacterias centrales o servidoras, el contenido de la memoria compartida, en cuanto a cantidad de AHL_g y AHL_r se refiere, cambia y las bacterias clientes actúan en consecuencia. Es importante notar que todas las bacterias centrales que forman la cepa de la capa de entrada actuarán al mismo tiempo debido a los umbrales progra-

mados en cada membrana, permitiendo así, la función síncrona de el biodispositivo diseñado.

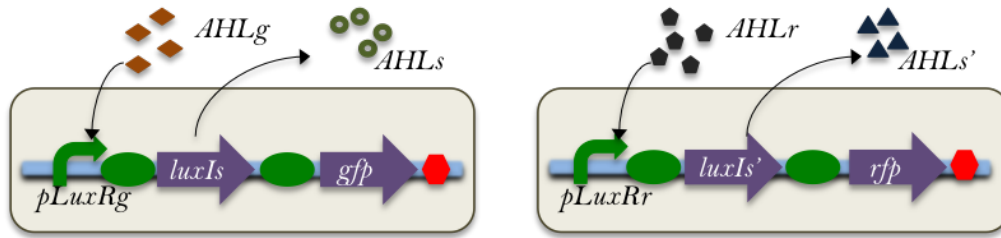


Figura 11.4: Diseño interno de las bacterias clientes

En la figura 11.4 se detalla la composición interna de las bacterias clientes. Es un diseño mucho más sencillo que el de la bacteria servidora ya que los clientes aquí no realizan tareas de sincronización. Por tanto, la funcionalidad del cliente se resume en realizar una acción determinada cuando se le otorga el turno de proceso.

Tan pronto como la bacteria verde (C_g , figura 11.4a), siente una concentración de moléculas AHL_g igual o superior al umbral diseñado para su cepa, activan la cascada de señales internas que derivan en la producción de moléculas AHL_s de fin de turno y la expresión de proteínas de fluorescencia verde gfp . Como consecuencia directa a la superación del umbral de C_g se activa el promotor inducible $pLuxRg$ que inicia el proceso de transcripción y traslación de los genes $luxIs$ y gfp . Como ya se ha dicho, el primero de dicho par de genes estructurales, se utiliza para producir moléculas de sincronización que serán situadas en la memoria compartida con la finalidad de notificar al servidor que la luz verde correspondiente a medio ciclo de oscilación (entendiendo como ciclo entero la transición de luz verde a luz roja) ha sido expresada satisfactoriamente por parte de la cepa de bacterias verdes y que el turno de proceso debe concederse al otro cliente. Es la bacteria servidora, como se ha explicado más arriba en esta sección, la encargada de cambiar el turno.

En la figura 11.4b) puede observarse el detalle de la bacteria roja C_r , que actúa de manera análoga a C_g durante la utilización de su turno de expresión para llevar a cabo la segunda mitad del ciclo de oscilación con la proteína de fluorescencia roja, rfp .

Cuadro 11.2: Reglas de inferencia de bajo nivel.

R1	if $ AHL_s \geq \delta_s$ then $pLuxRs = ACTIVE$
R2	if $pLuxRs = ACTIVE$ then $IPTG_s = TRUE \wedge CI434_s = TRUE$
R3	if $IPTG_s = TRUE$ then $pTac = ACTIVE$
R4	if $CI434_s = TRUE$ then $pSalCI434 = INACTIVE$
R5	if $pTac = ACTIVE$ then $luxIr_s = TRUE$
R6	if $ AHL_s' \geq \delta_s$ then $pLuxRs' = ACTIVE$
R7	if $pLuxRs' = ACTIVE$ then $sal_s = TRUE \wedge lac_s = TRUE$
R8	if $sal_s = TRUE$ then $pTac = INACTIVE$
R9	if $lac_s = TRUE$ then $pSalCI434 = ACTIVE$
R10	if $pSalCI434 = ACTIVE$ then $luxIg_s = TRUE$
R11	if $ AHL_g \geq \delta_{C_g}$ then $pLuxRg = ACTIVE$
R12	if $pLuxRg = ACTIVE$ then $luxIs_{C_g} = TRUE \wedge gfp_{C_g} = TRUE$
R13	if $ AHL_r \geq \delta_{C_r}$ then $pLuxRr = ACTIVE$
R14	if $pLuxRr = ACTIVE$ then $luxIs'_{C_r} = TRUE \wedge rf_{p_{C_r}} = TRUE$
R15	if $luxI_i = TRUE$ then $ AHL_i _t > AHL_i _{t-1}$
R16	if $luxI_i = FALSE$ then $ AHL_i _t < AHL_i _{t-1}$

El conjunto de reglas lógicas de inferencia de bajo nivel por el cual se gobierna el comportamiento oscilatorio se puede observar en la tabla 11.2. En el estado inicial, correspondiente al inicio de la computación, la memoria local propia de cada bacteria y la memoria compartida por las tres cepas están vacías. Así mismo, ninguno de los promotores inducibles está activo y por lo tanto los genes estructurales que dependen de la activación de éstos no están siendo expresados. Si se introduce manualmente en la comunidad heterogénea de bacterias el número de moléculas AHL_g suficiente par disparar la acción del umbral de las células C_g , la secuencia de reglas:

$$\{R11, R12(gfpon), R15, R1, R2, R3, R4, R16(gfpoff), R5, \\ R15, R13, R14(rfpon), R15, R6, R7, R8, R16(rfpoff), R9, R10, R15\} \quad (11.6)$$

será repetida indefinidamente.

A la vista de la secuencia 11.6, es importante resaltar la acción de sincronización que llevan a cabo dos reglas especiales y distintas a las demás como las R15 y R16. Estas dos reglas marcan el incremento y decremento, respectivamente, de moléculas AHL_i en la memoria compartida desde el instante anterior $t - 1$ al instante de tiempo actual t . La primera se ejecuta inmediatamente después de la activación de un gen de fluorescencia y la segunda marca la desactivación de éstos y la consecuente disminución de señal lumínica en la comunidad bacteriana.

Conviene notar que, debido nuevamente al carácter difuso de los procesos moleculares, es prácticamente imposible establecer un límite bien definido entre las dos señales lumínicas. Esto obliga la separación de las expresiones de *gfp* y *rfp* para evitar interferencias en el color de la población. El proceso de luminiscencia no es estrictamente digital, es decir, una bacteria pasa de estado *apagado* a *encendido* en un tiempo finito pero indefinido. Además, una vez reprimida la expresión de genes de fluorescencia, la bacteria tarda un tiempo en volver a su estado normal debido a que las proteínas de fluorescencia siguen activas.

Si estuvieran los eventos de expresión roja y verde muy juntos en el tiempo no se llegaría a distinguir claramente el color de la comunidad, permaneciendo éste en un tono intermedio. Para evitar este comportamiento indeseado, en la secuencia 11.6 se distinguen varias reglas situadas entre ambos procesos que, además de imprescindibles para el diseño, dejan espacio para la diferenciación del color.

11.4. Simulación

Como ya se ha detallado en las secciones previas de este capítulo, el biodespositivo a diseñar consiste en una comunidad bacteriana multi-cepa compuesta por tres componentes. Cada componente será una cepa modificada transgénicamente de manera diferente a las demás y diseñada para colaborar e interactuar con las otras de forma que sólo al mezclar las tres cepas pueda emerger el comportamiento buscado. Una vez los componentes están juntos, el dispositivo quedará armado pero inactivo. La activación del mismo debe producirse por la introducción en la comunidad del

número justo de moléculas AHL para disparar el umbral de las células servidoras. Es ésta una de las características más relevantes del biodispositivo: la activación externa.

Se podría haber realizado el diseño para que una vez unida, la comunidad multi-cepa estuviera ya activa y comenzara el comportamiento oscilatorio directamente sin la necesidad de activación. Sin embargo, se considera a esta arquitectura de más utilidad para futuros desarrollos en biodispositivos sintéticos.

Por ejemplo, el diseño de una comunidad celular encargada de buscar una toxina concreta (ya sea dentro de un cuerpo animal o en el agua) y reaccionar de manera definida al encontrarla, debe seguir este modelo de arquitectura con necesidad de activación externa. Tal sería el caso de los biosensores ya descritos previamente.

Para la simulación se ha hecho uso de la aplicación Simulink¹⁶ de Matlab y la librería BioBrick [Kid07] así como han sido necesaria la aplicación de varias ecuaciones y modelos matemáticos¹⁷ que simularan los procesos moleculares. También se han cotejado los resultados obtenidos en la herramienta *TinkerCell* [Cha09].

11.4.1. Construcción del modelo

En la figura 11.5 se puede observar el diseño del umbral situado a la entrada de cada bacteria emulando a la membrana celular. Esta membrana consta de tres componentes: un switch (*switch2* en la figura) de tres entradas y dos constantes (*Const5* y *Const6*) con valores 0 y 5 respectivamente. La entrada de la membrana corresponde al nivel de AHL exterior, en el caso de la figura 11.5 AHL_g , siendo la salida la conexión directa con el interior de la célula. El comportamiento de esta membrana está basado en el funcionamiento del switch, el cual devuelve como salida

¹⁶<http://www.mathworks.com/products/simulink/>

¹⁷La página web <http://openwetware.org> ofrece gran variedad de información sobre el comportamiento de procesos moleculares que han servido de punto de partida para la simulación computacional. El proyecto *OpenWetWare* representa una herramienta útil los entre investigadores para el intercambio de información.

el valor de la constante $Const6$ en caso de que el valor de la línea central de entrada al switch (nivel de AHL_g) supere un umbral prefijado, o el valor de la constante $Const5$ en caso contrario.

El valor de salida de la membrana determinará el carácter activo o pasivo de la bacteria en cuestión. Por ello la constante $Const5$ debe tener un valor igual a 0, con el fin de inhabilitar la actividad de la célula, así como la constante $Const6$ tendrá un valor positivo que iniciará los procesos celulares que la bacteria tenga diseñados internamente.

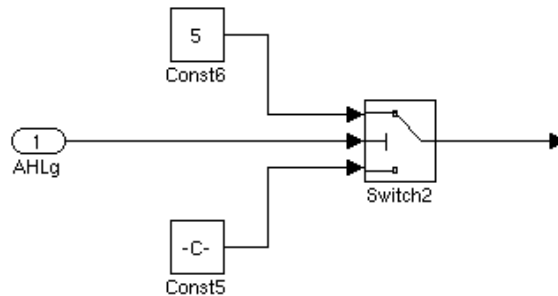


Figura 11.5: Modelado del umbral de sincronización

Durante la computación hay siempre una concentración variable en la memoria compartida de las moléculas que sirven como entrada a las bacterias servidoras, AHL_s y $AHL_{s'}$. Sin embargo, en el proceso de inicialización del dispositivo ambas concentraciones son nulas. Es ahí, donde la estructura de la figura 11.6 funciona para dar respuesta a esa activación manual que antes se comentaba. Dicha estructura representa la activación de la bacteria central mediante la introducción de un número elevado, con respecto al umbral, de moléculas AHL_s ($Const9$) que debe inicializar la cascada de señalización interna necesaria para conceder el turno de oscilación a la bacteria de fluorescencia roja. Para ello se comprueba previamente que ambas concentraciones son nulas antes de que el switch seleccione la entrada superior, lo cual simula la activación manual. Es importante puntualizar que este bloque se ejecutará únicamente al inicio de la computación, permaneciendo el resto del proceso oscilatorio inactivo.

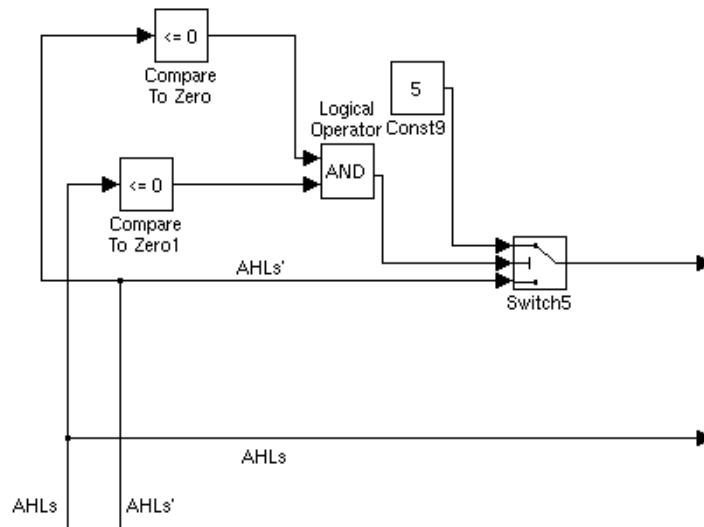


Figura 11.6: Modelado de la entrada manual

El modelo correspondiente al diseño de alto nivel puede observarse en la figura 11.7, donde se otorga relevancia a la disposición de las conexiones externas en detrimento de la estructura interna de los tres componentes del dispositivo. Aunque en la población todas las bacterias estarían *leyendo* en el medio nutriente todo tipo de moléculas constantemente, sólo reaccionarían a aquellas para las cuales han sido genéticamente modificadas. Por ese motivo, en la figura 11.7 sólo se detallan los procesos de lectura útiles para la comunidad. Así, la bacteria central está a la espera del desbordamiento del umbral por el incremento de moléculas AHL_s y $AHL_{s'}$ de fin de turno provenientes de los procesos de escritura (entendiendo por escritura la producción de moléculas y su lanzamiento al exterior de la membrana celular) de las bacterias verde y roja respectivamente. Del mismo modo las bacterias clientes reciben el testigo de inicio de turno, AHL_g o AHL_r dependiendo del ciclo de oscilación, que proviene de la bacteria central.

La fluorescencia que se produce en las bacterias clientes se monitoriza para comprobar los resultados de la oscilación. Aunque esta expresión lumínica se representa en la figura 11.7 como si fueran salidas de dichas bacterias, la producción de proteínas de fluorescencia se lleva a cabo en el ámbito de la memoria local de cada célula. Es decir, las proteínas *gfp* y *rfp* no traspasan la membrana de las bacterias

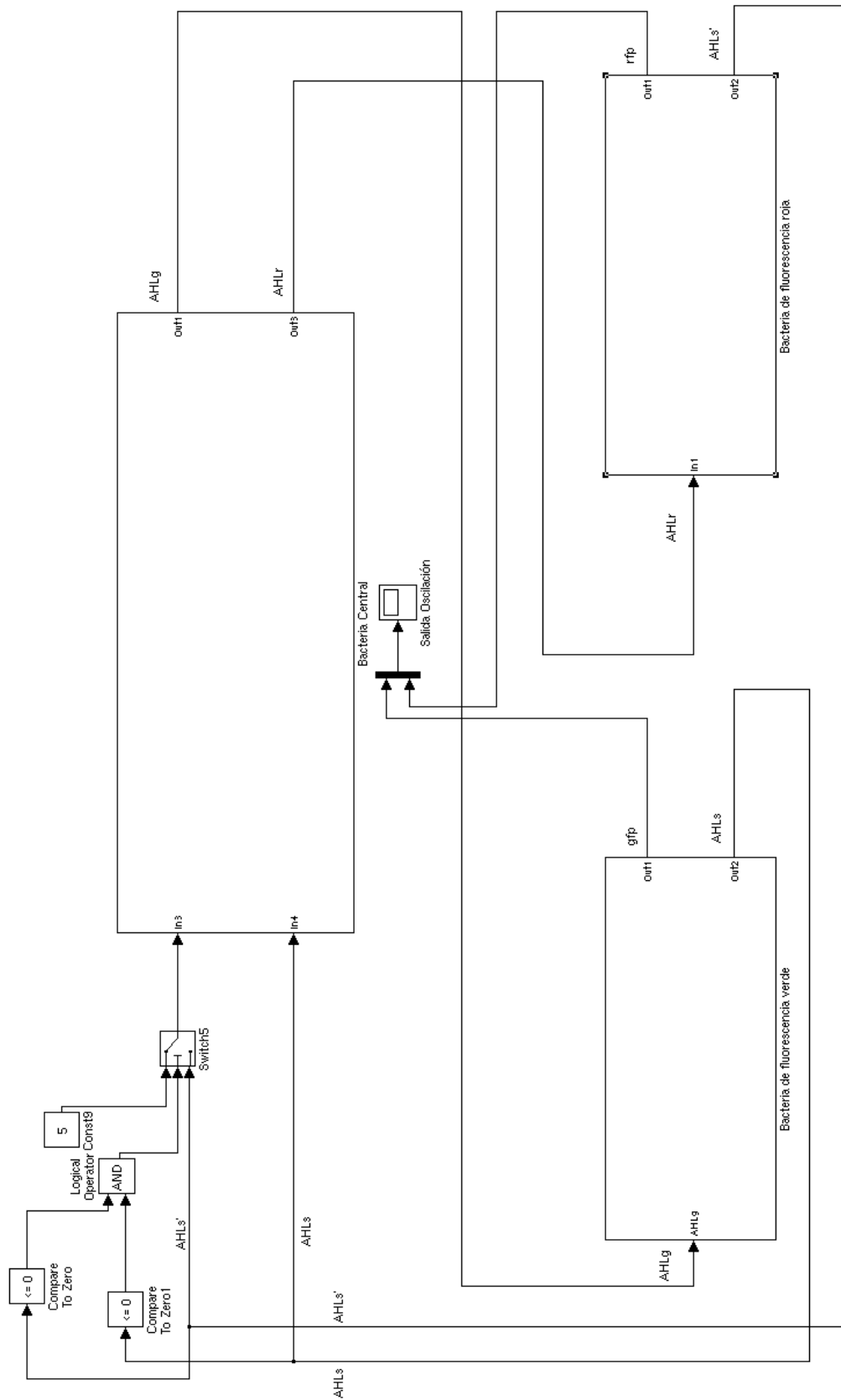


Figura 11.7: Diseño de alto nivel del oscilador

que las contienen.

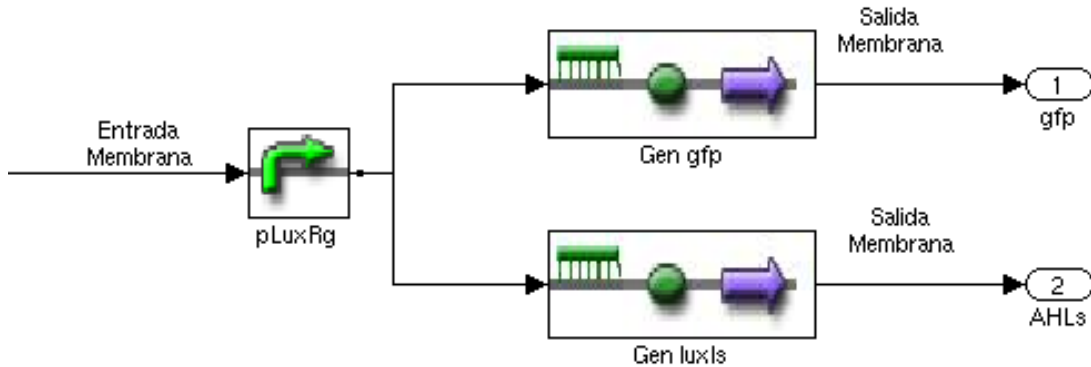


Figura 11.8: Diseño de la bacteria verde

En cuanto al diseño de bajo nivel, las dos cepas clientes son modificadas siguiendo una estructura interna similar en la que únicamente cambia la naturaleza de los genes y promotores involucrados en ella. En la figura 11.8 puede observarse dicha estructura en el caso de las bacterias de fluorescencia verde. Este sencillo diseño hace uso de un promotor específico para la molécula de concesión de turno que ha de leer y dos genes estructurales cuya expresión depende del promotor.

En el caso de la figura los genes son *gfp* y *luxIs*, siendo *rfp* y *luxIs'* en el diseño de clientes rojos. Aunque *in-vivo* los genes expresados por un mismo promotor están situados en serie en una misma cadena de ADN aquí se ha optado por su situación en paralelo ya que esta paralelización simplifica el modelo y no influye en los resultados de la simulación. Como se ha comentado previamente, el motivo de conectar la salida del gen estructural *gfp* con el exterior de la membrana es únicamente la monitorización de la expresión genética de éste.

El promotor *pLuxRg* de la figura 11.8 es un promotor inducible que actúa como un valor constante que se activa o desactiva dependiendo de la señal que reciba de la membrana. Sin embargo el diseño del gen estructural es bastante más complejo y requiere particular atención.

En la figura 11.9 se detalla la secuencia de acciones que lleva al gen estructural *gfp* a expresar su información lumínica. Primero, el RNA mensajero aplica una

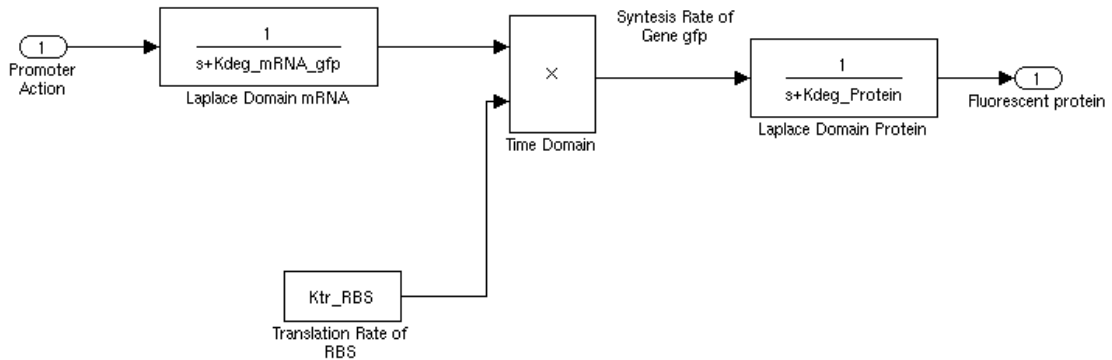


Figura 11.9: Detalle de la expresión génica

ecuación de Laplace al valor de salida del promotor. A continuación, el valor dado al RBS del gen amplifica o disminuye el valor de salida del RNA mensajero y, por último, se aplica otra ecuación de Laplace correspondiente al valor de la variable *Kdeg_Protein* a la tasa de síntesis genética. Como salida de este proceso secuencial, se obtiene la expresión del gen estructural.

La figura 11.10 muestra el esquema de bloques que simula la acción de la bacteria servidora. Los bloques denominados *Umbral de Membrana* representan el esquema de la figura 11.5 y los bloques denominados *Operador* simulan la acción de los distintos operadores de los promotores híbridos. Estos promotores tienen dos operadores o lugares de unión diferenciados para las moléculas inductoras y las represoras. El estado de ambos operadores se combina para dar lugar al valor final que saldrá del promotor (potencia de transcripción de éste).

De esta manera cada bloque *Operador*, detallado en la figura *fig:hyboperator*, resta el valor de la señal represora al valor de la inductora. Ya que la señal represora tiene que funcionar de manera estrictamente bloqueante se dobla su valor para simular esta interacción.

En el interior del bloque *Operador* se mantiene un control exhaustivo del valor de la señal final con la que el promotor debe transcribir el gen objetivo. Así, será éste siempre mayor o igual a 0 sin poder mantener un valor negativo.

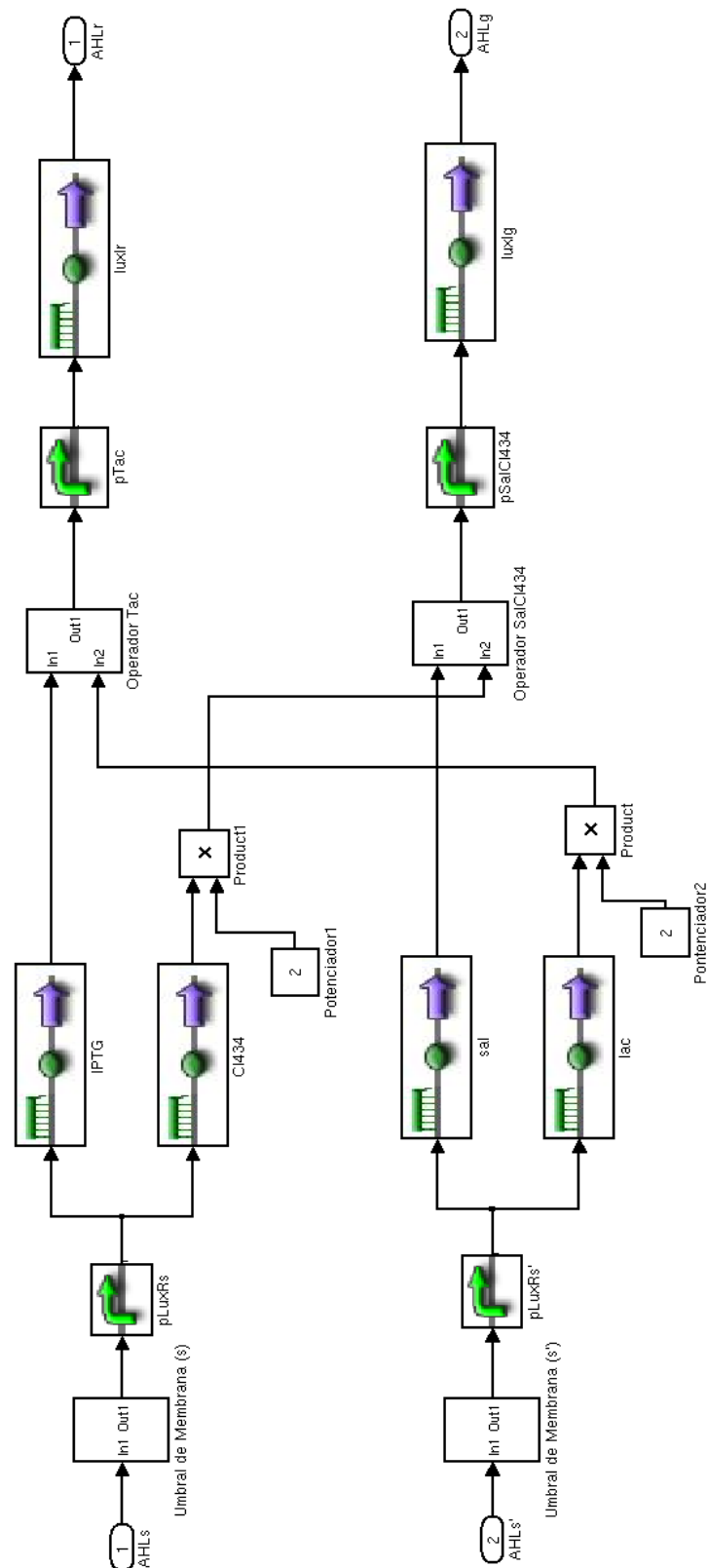


Figura 11.10: Diseño de la bacteria central

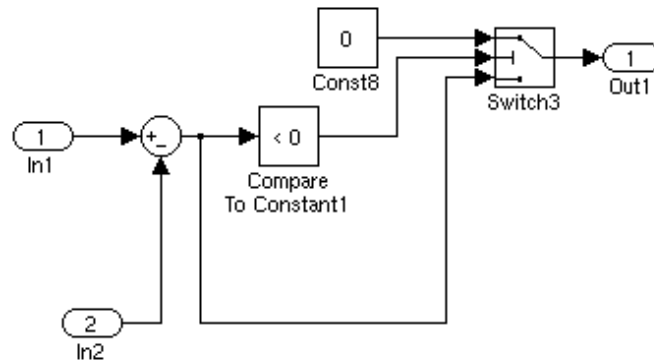


Figura 11.11: Simulación de los operadores del promotor híbrido.

11.4.2. Resultados

11.4.2.1. Simulación de componentes

Cada componente del sistema, la bacteria servidora y las clientes, son simuladas para comprender su comportamiento por separado antes de realizar las pruebas globales del sistema.

Cuadro 11.3: Tabla de verdad de la bacteria servidora. S: variable binaria que indica si la bacteria tiene salida o no.

AHLs	AHLs'	AHLg	AHLr	S
0	0	0	0	0
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

La tabla 11.3 muestra el comportamiento de la bacteria servidora. Es importante hacer hincapié en el cuarto estado de la tabla, ya que éste guiará el comportamiento del sistema en buena medida. Así se observa que al recibir ambos estímulos simultáneamente la bacteria se bloquea gracias a las señales represoras y no produce salida alguna. Hay que notar que el valor binario de cada variable hace referencia a la superación o no del umbral por parte de un determinado tipo de moléculas.

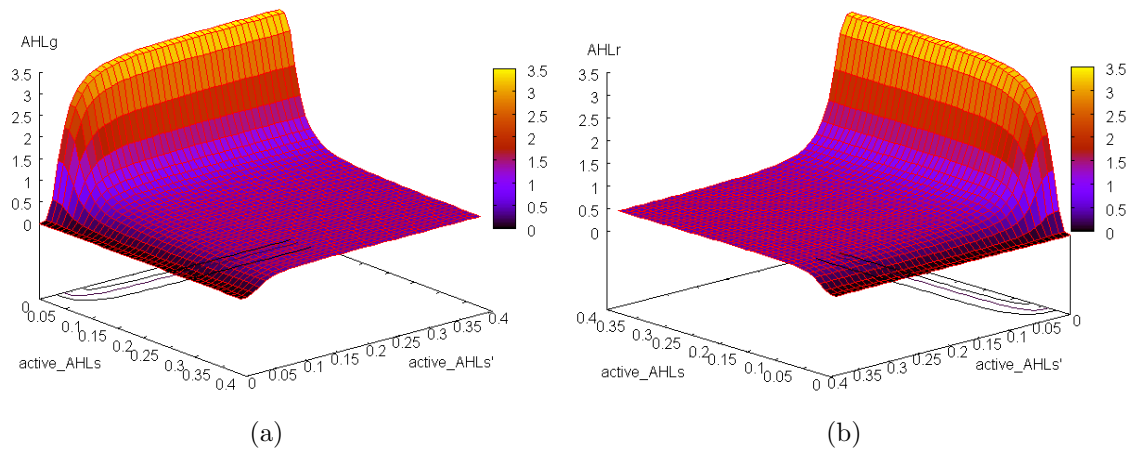


Figura 11.12: Comportamiento de la bacteria servidora en función de las entradas. a) producción de AHLg. b) producción de AHLr.

La figura 11.12 simula analógicamente (por solución en lugar del carácter digital estricto de la tabla de verdad) la actividad de la bacteria servidora. Se observa que la salida esperada se obtiene cuando la molécula inductora se encuentra presente y la represora es prácticamente nula.

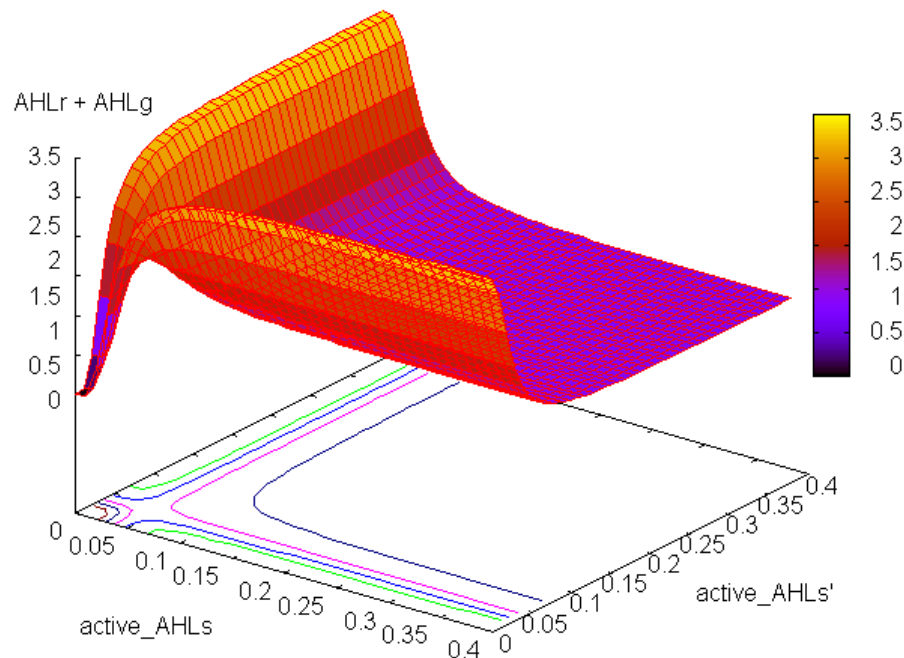


Figura 11.13: Producción total de la bacteria servidora sin distinción de la naturaleza de las moléculas de salida.

La fuerte regulación del promotor híbrido se hace patente al incrementarse el valor de la fuerza represora, en cuyo caso es indiferente la concentración de las moléculas inductoras ya que el promotor se encuentra en estado de *bloqueo*. Esto hace que la interferencia entre las dos salidas de la bacteria servidora se reduzca al mínimo y el ruido genético sea así controlado por este cruzamiento de fuerzas.

La figura 11.13 ilustra este comportamiento en una gráfica que es la unión de las dos gráficas de la figura 11.12. Se puede observar como la intersección es realmente mínima.

Con el fin de comparar el resultado de reducción de ruido provocado por los promotores híbridos se realiza también una simulación con un diseño alterado de la bacteria servidora en el que se omiten las fuerzas represoras de las moléculas *Cl434* y *lac* y se sustituyen los promotores híbridos por simples promotores inducibles. El estado 1 – 1 en la entrada de la bacteria servidora provoca entonces un 1 – 1 en la salida en diferencia al 0 – 0 provocado en el diseño original.

Las gráficas de la figura 11.14 muestran el comportamiento de la bacteria con la nueva programación.

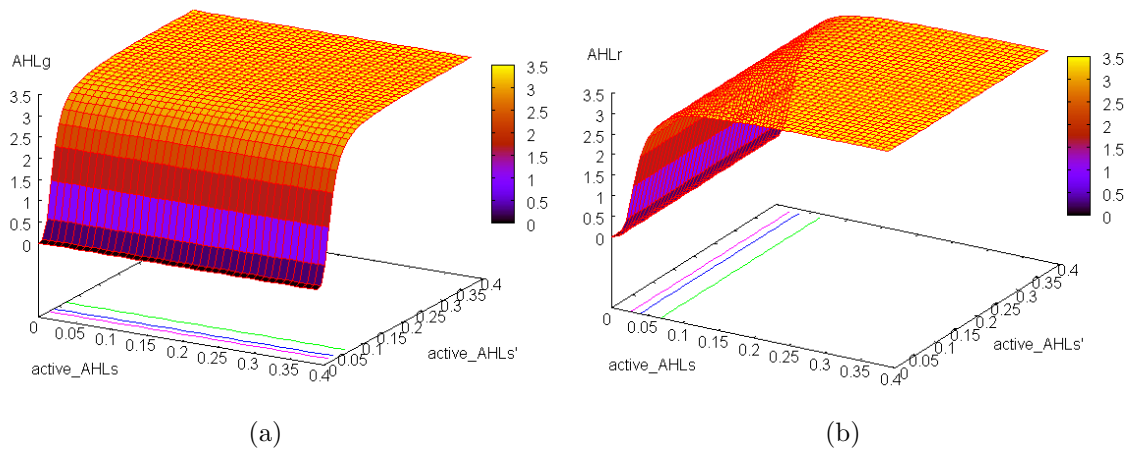


Figura 11.14: Comportamiento de la bacteria servidora con programación primaria en función de las entradas. a) producción de AHLg. b) producción de AHLr

La unión de las funciones de salida de AHL_r y AHL_g para esta nueva configuración se muestra en la figura 11.15 donde se observa como la interferencia entre

ambas señales es considerablemente extensa.

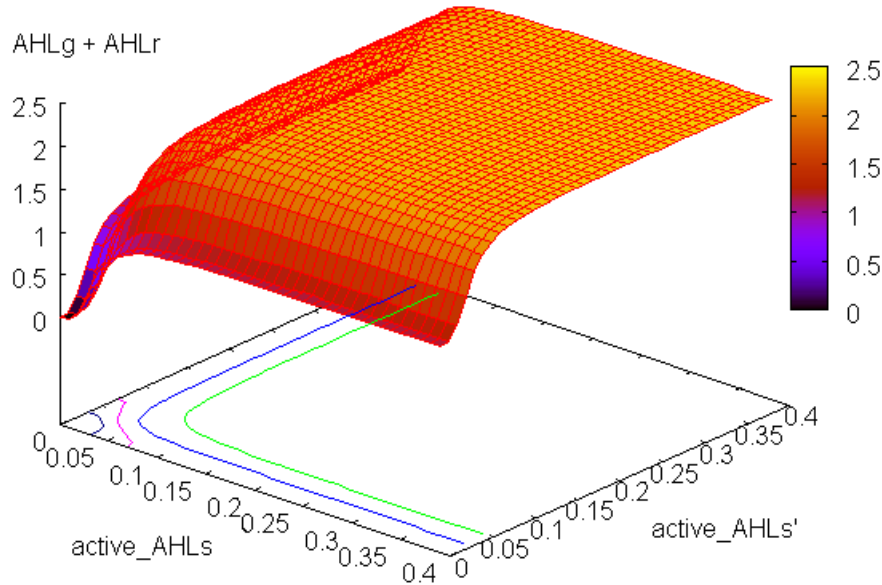


Figura 11.15: Producción total de la bacteria servidora con programación sencilla sin distinción de la naturaleza de las moléculas de salida.

La reducción del ruido es por tanto uno de los grandes logros que el empleo de promotores híbridos consigue establecer en el diseño del biodispositivo oscilatorio que ocupa el desarrollo de este capítulo.

Por último, la figura 11.16 muestra el comportamiento de los clientes para todos los posibles valores de entrada. Cabe destacar que, debido al diseño de su estructura interna, la bacteria cliente verde muestra un comportamiento indiferente a la cantidad de AHL_r que haya en el entorno así como ocurre de igual manera en el caso de las bacterias clientes rojas y la concentración de moléculas AHL_g .

11.4.2.2. Simulación del sistema global

Cuatro simulaciones son llevadas a cabo en situaciones diferentes con el fin de establecer conclusiones relevantes y realizar un estudio del comportamiento oscilatorio en profundidad. Las simulaciones se pueden agrupar en dos subgrupos:

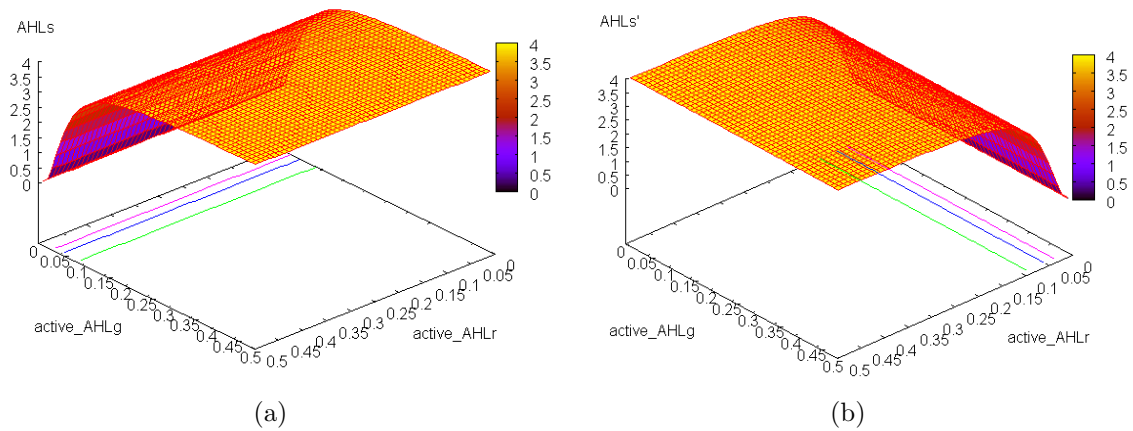


Figura 11.16: Clients behaviour. a) Green client. b) Red client.

- Simulaciones estáticas.

Se consideran aquí las poblaciones bacterianas correspondientes a las tres cepas como conjuntos finitos y fijos en el tiempo. Es decir, el número de agentes que componen las cepas resulta invariable durante la simulación. Se evalúan dos modelos estáticos:

- ▷ Diseño sin promotores híbridos. Esta simulación representa el diseño más intuitivo de la bacteria central en la que se omiten las fuerzas represoras de las moléculas *CI434* y *lac* así como se sustituyen los promotores híbridos por simples promotores inducibles.
- ▷ Diseño con promotores híbridos. Esta simulación representa el diseño exacto de la figura 11.3 y que ya ha sido detallado en el apartado anterior. La comparación de los resultados obtenidos de la simulación correspondiente al diseño sin promotores híbridos con los de este modelo, permitirán evaluar la eficacia de estos promotores en cuanto a tareas de sincronización y reducción del ruido genético.

- Simulaciones dinámicas.

Se considera aquí la velocidad de crecimiento de las poblaciones celulares que forman el biodispositivo. Las dos simulaciones dinámicas son llevadas a cabo

con el modelo que incluye los promotores híbridos. Estos dos casos son:

- ▷ Diseño con función de crecimiento común.

Para el desarrollo de esta simulación, se supone que las tres cepas que componen el biodispositivo presentan exactamente el mismo crecimiento. Éste es ascendente, estudiándose así el comportamiento oscilatorio cuando las cepas exceden un determinado tamaño.

- ▷ Diseño con función de crecimiento condicionada.

Puesto que es prácticamente imposible encontrar en la naturaleza comunidades heterogéneas de bacterias en las que las distintas especies se comporten según la misma función de crecimiento, en esta última simulación se utilizan distintas funciones para cada cepa. Estas funciones de crecimiento no son independientes entre sí, sino que presentan una fuerte dependencia simulando una situación de competición entre las tres comunidades. Se obtienen resultados del comportamiento oscilatorio cuando una o más cepas desaparecen.

Los resultados obtenidos de la simulación correspondiente al diseño sin promotores híbridos pueden observarse en la gráfica de la figura 11.17. Al no considerarse función de crecimiento, se consideran las cepas como poblaciones constantes en el tiempo. Al desaparecer en esta simulación las fuerzas represoras sobre los promotores híbridos, convirtiéndolos en promotores inducibles, se observa como el ruido genético estocástico en el interior de la bacteria central resulta un grave problema. Las concentraciones inapropiadas de moléculas inductoras convierten el proceso de selección de umbral en una tarea realmente complicada.

Aunque el comportamiento oscilatorio de la comunidad se alcanza satisfactoriamente, tal y como se ve en la figura 11.17, la principal conclusión de esta primera simulación se centra en la viabilidad del diseño *in-vivo*.

Considerando que la concentración mínima, c_0 , de moléculas AHL que produce una cepa de la población es igual a 0 y la concentración máxima, c_m , es igual a 5 tendremos concentraciones en un intervalo $[0 \cdots 5]$ variable en el tiempo para cada tipo

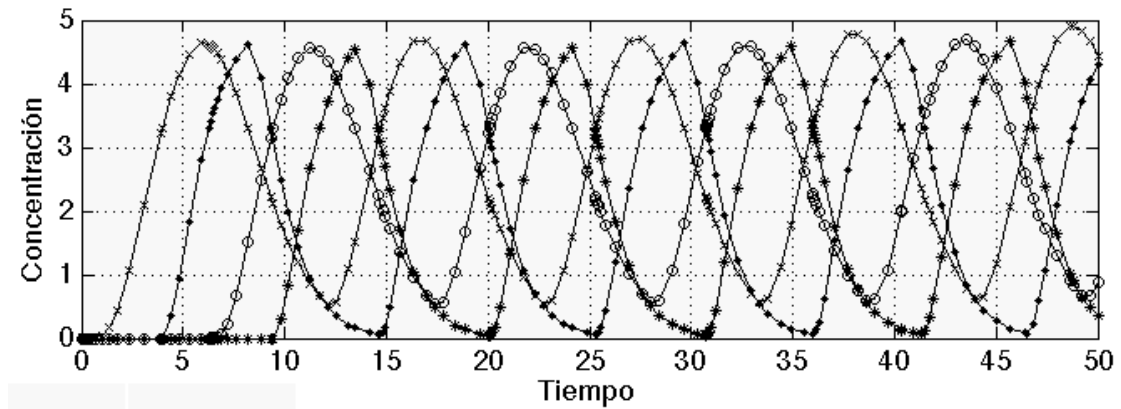


Figura 11.17: Simulación estática sin promotores híbridos. La oscilación mide la concentración de las distintas moléculas AHL en solución: AHLr(\times), AHLg(\circ), AHLs'(\bullet), AHLs($*$). La fluorescencia verde coincide con la expresión de AHLs y la roja con AHLs'.

de molécula. En este escenario, el umbral diseñado en la membrana de cada bacteria debe tener un valor perteneciente al intervalo $[3, 2 \dots 3, 3]$ para la correcta ejecución del biodispositivo que, de otro modo, no alcanzará el comportamiento oscilatorio. Este último intervalo representa únicamente el 2% del intervalo de salida de una cepa, lo cual da una clara idea de lo preciso que debe ser el diseño del umbral en este caso. Obviamente, esta precisión necesaria representa una importante desventaja a la hora de protocolizar el diseño de la oscilación *in-vivo* ya que la manipulación de seres vivos resulta frecuentemente en resultados inesperados y los parámetros del diseño deben ser lo más generalizables, maleables y permisibles posible.

Dicho problema es debido a que el ruido genético en la bacteria central no está siendo controlado de ninguna forma. Para ello, se introducen los promotores híbridos y las fuerzas de represión provenientes de la expresión de las moléculas represoras por parte de los genes *CI434* y *lac*, simulando así el diseño detallado en secciones previas. La figura 11.18 muestra la oscilación resultante de esta segunda simulación.

Las diferencias que se observan en la figura 11.18 con respecto a la simulación anterior son muy leves y casi imperceptibles. Sin embargo, la importancia de ésta última radica en la eliminación del ruido genético que se hace notar a la hora de di-

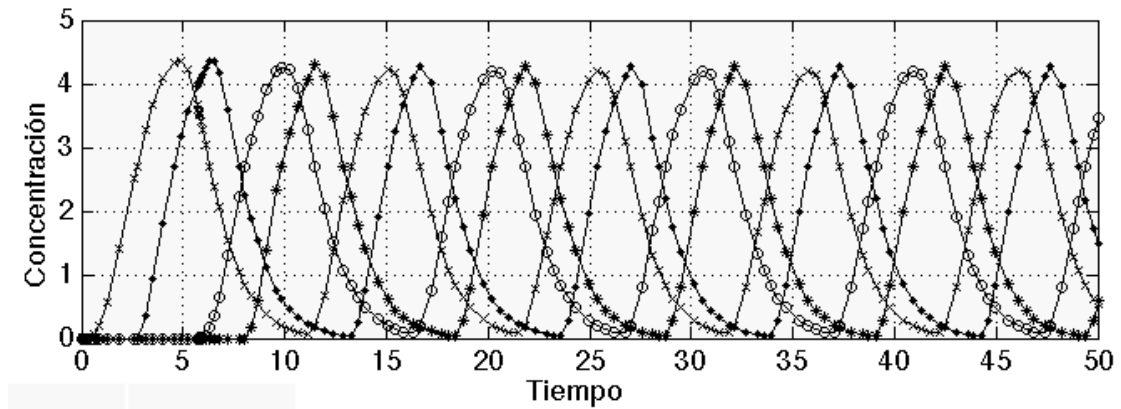


Figura 11.18: Simulación estática con promotores híbridos y umbral alto. La oscilación mide la concentración de las distintas moléculas AHL en solución: AHLr(\times), AHLg(\circ), AHLs'(\bullet), AHLs($*$). La fluorescencia verde coincide con la expresión de AHLs y la roja con AHLs'.

señalar los umbrales de oscilación. En esta ocasión, el umbral diseñado en la membrana de cada célula debe tener un valor perteneciente al intervalo $[\simeq 0, 1 \cdots 3, 1]$.

Considerando nuevamente un valor máximo de 5 para representar la salida total producida por una cepa, éste intervalo representa el 60 % del rango de salida. Esto significa que la regulación producida por las fuerzas represoras sobre los promotores híbridos permite ampliar considerablemente el margen de error en el diseño del umbral, otorgando una mayor viabilidad al modelo. Así, cuanto mayor sea el margen de configuración de los distintos componentes del sistema, mayor será también la plausibilidad de éste.

Debido al gran espacio de definición de umbral en la membrana, se producen resultados distintos en la simulación dependiendo de el valor de éste. Así la figura 11.19 muestra la simulación en situación de bajo umbral. Como puede observarse la oscilación se consigue aunque la frecuencia disminuye considerablemente.

Esto es debido a la superación simultánea de los umbrales de las moléculas de sincronización (s y s') en las bacterias servidoras. Debido a su programación, este estado provoca la *espera* del sistema hasta que una de las dos señales baje y el ciclo de oscilación vuelva a comenzar.

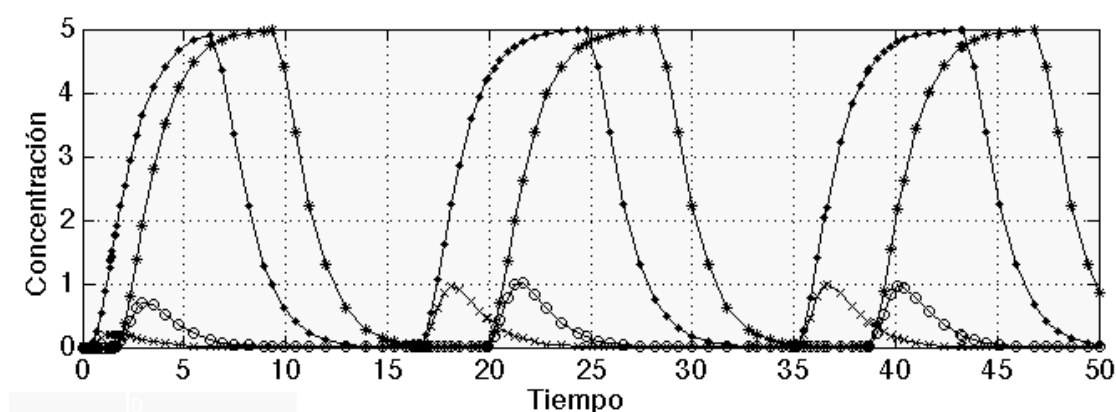


Figura 11.19: Simulación estática con promotores híbridos y umbral bajo. La oscilación mide la concentración de las distintas moléculas AHL en solución: AHLr(\times), AHLg(\circ), AHLs'(\bullet), AHLs($*$). La fluorescencia verde coincide con la expresión de AHLs y la roja con AHLs'.

Una vez estudiado el diseño del biodispositivo considerando poblaciones estáticas de bacterias y habiendo extraído resultados positivos de la utilización de los promotores híbridos, se pasa a comprobar el funcionamiento del oscilador en situaciones más realistas de comunidades dinámicas.

El crecimiento de poblaciones bacterianas hace referencia a un proceso por el cual se realiza un incremento ordenado de estructuras y componentes celulares. Siendo éste un tema extenso y complejo, se describen a continuación las principales características de este crecimiento ¹⁸ para finalizar el capítulo con las dos últimas simulaciones.

Son muchos los factores que determinan la función de crecimiento de una especie bacteriana por lo que habitualmente resulta complicado extraer ecuaciones precisas que muestren un comportamiento realista para un determinado instante.

Así, el crecimiento depende en última instancia de la habilidad de la célula para crear nuevos componentes sirviéndose de los nutrientes que encuentre disponibles en

¹⁸Para más información, visitar el *Todar's Online Textbook of Bacteriology* en la web <http://www.textbookofbacteriology.net> donde se puede encontrar información detallada sobre las bacterias y su ciclo de vida.

su entorno. En la mayoría de las bacterias, el crecimiento implica el aumento de la masa celular y del número de ribosomas, la duplicación del cromosoma bacteriano, la síntesis de una nueva membrana y plasma celular, división celular, etc., en lo que resulta un proceso de reproducción llamado fisión binaria. En este proceso, el intervalo de tiempo requerido para la división de una bacteria o el intervalo de tiempo requerido para la duplicación de una población se llama tiempo de generación.

Como veremos a continuación, los tiempos de generación de las distintas especies en la naturaleza oscilan entre pocos minutos y varios días dependiendo no sólo de la especie sino también de las condiciones específicas. En el caso de seres unicelulares como las bacterias, el crecimiento celular de una población puede medirse con dos parámetros: la masa celular y el número de células. En lo que concierne a la simulación del biodispositivo diseñado en este capítulo sólo se ha tenido en cuenta el número de bacterias en la población, siendo el primer parámetro de escasa relevancia en este ejemplo.

El crecimiento bacteriano no responde a una función estrictamente creciente, sino que viene definido por una curva semejante a la campana de Gauss. La definición anterior de tiempo de generación hace referencia a una fase de la curva denominada fase de crecimiento exponencial, durante la cual la población se duplica a intervalos relativamente uniformes siguiendo la progresión geométrica $\{1, 2, 4, 8, 16, \dots\}$ o, representado de otra forma, $\{2_0, 2_1, 2_2, 2_3, \dots, 2_n\}$ donde n es el número de generaciones. Para la duplicación exacta de la población durante esta fase del ciclo de vida bacteriano se suponen condiciones óptimas.

Casi siempre se reconocen las mismas cuatro fases en la curva de crecimiento de una población bacteriana cultivada en un sistema cerrado:

- Fase inicial.

Esta fase ocurre inmediatamente después de la introducción de las bacterias en el medio nutriente y, durante su desarrollo, la población presenta un crecimiento prácticamente nulo. Esto se debe a que las células durante esta fase realizan tareas de adaptación al nuevo medio tales como aumentar en tamaño,

sintetizar enzimas, incrementar la actividad metabólica, etc. La dificultad de estas últimas tareas determinará la duración de la fase inicial.

- Fase exponencial.

Esta es la fase a la que se hacía referencia anteriormente y en la que la división celular alcanza su máxima actividad siguiendo progresiones geométricas exponenciales. El periodo de duplicación o tiempo de generación bacteriana depende, entre otros factores, de la composición del medio nutriente y las condiciones de incubación de la población.

- Fase estacional.

En esta fase la curva vuelve a presentar un crecimiento nulo ya que la fase anterior no puede continuar indefinidamente. Esto puede ser debido a dos factores: 1) El número tan elevado de bacterias en el medio nutriente hace que se agoten rápidamente los recursos de éste para mantener más seres vivos. 2) El espacio del medio de cultivo (placa de petri, tubo de ensayo, etc.) ya no puede dar cabida a más bacterias. Cualquiera de los dos factores anteriores provoca un estancamiento del número de agentes en la población.

- Fase de muerte.

Cuando la fase estacional continúa, se alcanza la última fase en el ciclo de vida de las bacterias. Durante esta fase la población de células decrece geométricamente siguiendo aproximadamente el proceso inverso a la fase exponencial.

Los tiempos de generación de algunas especies bacterianas pueden leerse en la tabla 11.4. Como ya se ha dicho, el tiempo de generación es muy variable dependiendo de las condiciones de cultivo incluso en la misma especie. Por ejemplo, el tiempo de generación de la bacteria *E. coli* en laboratorio es de 15-20 minutos mientras que la misma bacteria situada en el tracto intestinal muestra un tiempo de generación de entre 12 y 24 horas.

Para probar el comportamiento oscilatorio en una población con crecimiento real, hay que modelar la curva de crecimiento bacteriano [Put84] en la correspondien-

Cuadro 11.4: Tiempos de generación en diferentes especies de bacterias bajo condiciones de cultivo óptimas.

Especie	Medio nutriente	Tiempo de generación (min.)
<i>Escherichia coli</i>	Glucosa-sales	17
<i>Bacillus megaterium</i>	Sacarosa-sales	25
<i>Streptococcus lactis</i>	Leche	26
<i>Streptococcus lactis</i>	Lactosa	48
<i>Staphylococcus aureus</i>	Heart infusion broth	27-30
<i>Lactobacillus acidophilus</i>	Leche	66-87
<i>Rhizobium japonicum</i>	Mannitol-sales	344-461
<i>Mycobacterium tuberculosis</i>	Sintético	792-932
<i>Treponema pallidum</i>	Rabbit testes	1980

te ecuación matemática de crecimiento. Teniendo en cuenta la progresión geométrica de división celular, podríamos enunciar la ecuación de crecimiento de la población de la siguiente forma:

$$\frac{dN}{dt} = r \bullet N \quad (11.7)$$

donde N es el número de individuos que forman la población a estudio, t el tiempo y r la tasa intrínseca de incremento natural. El parámetro r se obtiene de restar la tasa de muerte celular a la tasa de nacimiento.

Como resultado de la ecuación 11.7 se obtiene una gráfica exponencial sin un fin definido, lo cual no representa el crecimiento de una población en cultivo cerrado. Obviamente, la población no podrá expandirse para siempre y existirá un límite al número de individuos que la comunidad puede albergar. Ese límite es conocido como capacidad de almacenamiento. El comportamiento que, así, deriva en la fase estacional del ciclo de vida poblacional se consigue añadiendo el siguiente término a la ecuación 11.7:

$$\left(1 - \frac{N}{k}\right) \quad (11.8)$$

donde k es la capacidad de almacenamiento.

Es importante hacer hincapié en la relación entre los parámetros k y N ya que son éstos los que marcarán los ciclos de vida poblacionales. Así cuando $N \ll k$ la comunidad de bacterias se dirige hacia la fase exponencial, mientras que en el caso de ser $N = k$ se elimina la posibilidad de futuros crecimientos y la última fase empieza a ejecutarse.

Uniendo las ecuaciones 11.7 y 11.8 para formar la ecuación de crecimiento definitiva se obtiene:

$$\frac{dN}{dt} = r \bullet N \bullet \left(1 - \frac{N}{k}\right) \quad (11.9)$$

En contraposición a las dos primeras simulaciones, que se llevaron a cabo durante la fase inicial del ciclo de vida bacteriano en el que el crecimiento es nulo¹⁹, en la tercera simulación se estudia el comportamiento oscilatorio durante la fase exponencial con una particularidad: todas las cepas presentan exactamente la misma función de crecimiento.

Para obtener una curva parcial de crecimiento (se desea probar el oscilador en fase de crecimiento) se desarrolla una ecuación diferencial que de respuesta a un simple modelo de crecimiento bacteriano. Se mantiene el carácter atemporal del eje de abscisas y que el objetivo de esta simulación es la comprensión del mecanismo en detrimento de las mediciones temporales referidas a una especie bacteriana concreta en una situación específica. Se considera proporcional el incremento del número de bacterias que nacen con respecto a las bacterias existentes y el número de bacterias que mueren proporcional al número de bacterias existentes al cuadrado:

$$\frac{dx}{dt} = bx - px^2 \quad (11.10)$$

donde x es el número de bacterias existentes en la población, bx el crecimiento de nacimientos, px^2 el crecimiento de muertes y las constantes b y p hacen referencia a la relación de bacterias por hora, siendo 1 y 0,5 respectivamente.

¹⁹Aunque en estas primeras simulaciones no se considera el crecimiento bacteriano, se continúa simulando la realidad y no un caso imaginario. Estas simulaciones corresponderían, por tanto, a cualquiera de las dos fases estacionarias del ciclo de vida de las bacterias cultivadas en un medio cerrado.

El diagrama de Simulink que refleja el comportamiento de la ecuación 11.10 se puede observar en la figura 11.20.

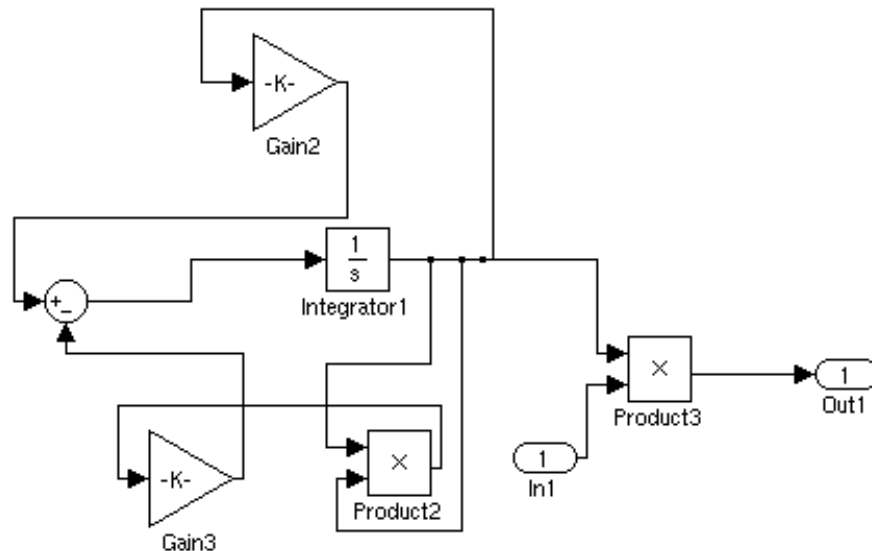


Figura 11.20: Bloque para simular de forma parcial la fase exponencial del ciclo de vida bacteriano. El número inicial de bacterias en la población se especifica en el bloque *integrator*.

La figura 11.21 muestra la simulación del oscilador en situación de crecimiento. Se observa cómo los niveles moleculares son cada vez mayores al aumentar la población celular. Debido al proceder del servidor en aquel estado en el que ambos umbrales se superan simultáneamente, el comportamiento oscilatorio no se ve afectado por el crecimiento. Este hecho prueba una característica fundamental del biodispositivo: la resistencia al crecimiento celular y consecuente robustez en el diseño. Se estudian a continuación los resultados obtenidos al no utilizar las fuerzas represoras.

Los resultados derivados de la simulación mostrada en la figura 11.22 corresponden a una simulación del oscilador en situación de crecimiento y con el diseño primario (sin promotores híbridos) en la bacteria servidora. Éstos concluyen que existe un determinado momento durante la fase exponencial del ciclo de vida en el

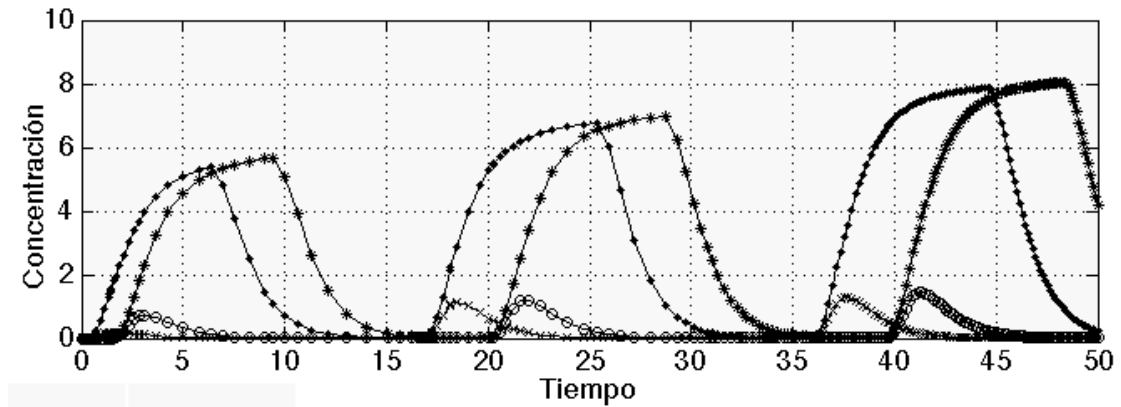


Figura 11.21: Simulación dinámica con función de crecimiento común. Diseño original con promotores híbridos. La oscilación mide la concentración de las distintas moléculas AHL en solución: AHLr(\times), AHLg(\circ), AHLs'(\bullet), AHLs($*$). La fluorescencia verde coincide con la expresión de AHLs y la roja con AHLs'.

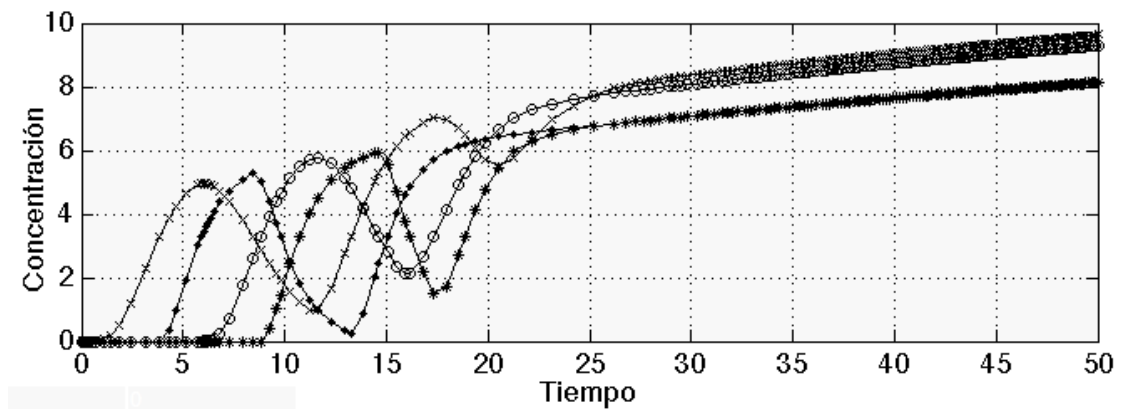


Figura 11.22: Simulación dinámica con función de crecimiento común. Diseño primario sin promotores híbridos. La oscilación mide la concentración de las distintas moléculas AHL en solución: AHLr(\times), AHLg(\circ), AHLs'(\bullet), AHLs($*$). La fluorescencia verde coincide con la expresión de AHLs y la roja con AHLs'.

que las moléculas de AHL necesarias para la correcta sincronización son tan numerosas, debido al aumento de su producción proporcional al aumento de bacterias, que se produce una sobrecarga en el biodispositivo.

Efectivamente, al haber un número exponencial de moléculas AHL no hay tiempo a que estas desaparezcan a tiempo y se acumularán en el medio nutriente.

Como resultado los umbrales de todas las células se encuentran excedidos constantemente y, debido a la actividad de la bacteria servidora sin fuerzas represoras, ambas luces se expresan permanentemente.

La figura 11.22 ilustra el comportamiento oscilatorio durante esta fase hasta la sobrecarga y desbordamiento de los umbrales. A diferencia del modelo original, la falta de utilización de señales represoras lleva a la terminación del comportamiento oscilatorio.

Una interesante simulación del modelo original se observa en la figura 11.23 en la que se ha modificado la tasa de degradación, hasta ahora igual al resto, de las moléculas AHLs'.

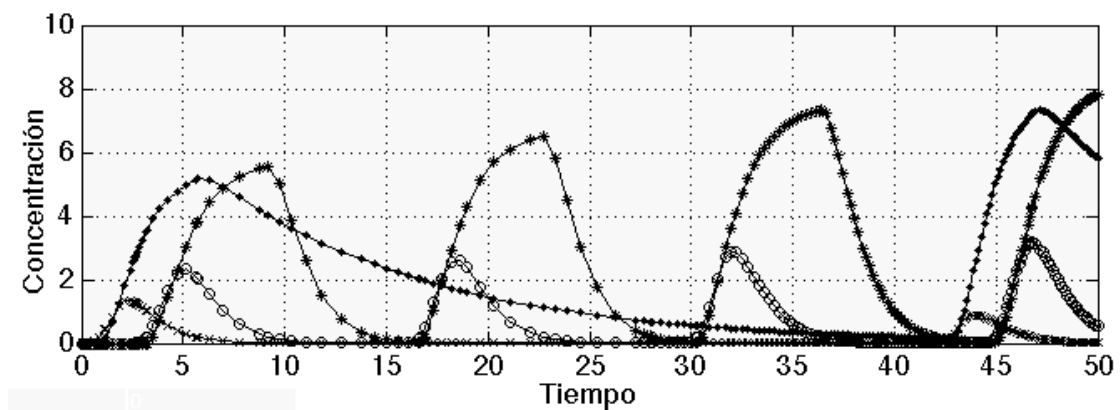


Figura 11.23: Simulación del sistema original con degradación lenta de moléculas AHLs'. La oscilación mide la concentración de las distintas moléculas AHL en solución: AHLr(\times), AHLg(\circ), AHLs'(\bullet), AHLs(\ast). La fluorescencia verde coincide con la expresión de AHLs y la roja con AHLs'.

Se prueba así nuevamente la robustez en el diseño ya que la oscilación no termina sino que se altera y se adapta el patrón lumínico de la oscilación. Si antes sucedía la fluorescencia verde a cada roja, ahora suceden cuatro ciclos de oscilación verde a cada evento fluorescente rojo.

La última simulación a realizar, prueba el comportamiento oscilatorio en una situación de competición entre las bacterias de la comunidad heterogénea. Es éste

un escenario más realista al modelar las interacciones entre cepas de manera que cada una seguirá su propia función de crecimiento dependiendo de lo que hagan las demás cepas en un momento determinado. La competición intercelular se produce como consecuencia de estar las bacterias obligadas a compartir los recursos de un espacio cerrado y limitado. Surge entonces una lucha por los nutrientes y el espacio que deriva en la regulación de poblaciones, la extinción de cepas e incluso la creación de nuevas especies bacterianas que respondan a las nuevas necesidades del medio.

Para simular esta competición, hay que añadir a la ecuación de crecimiento de una cepa 11.9 un nuevo término que haga referencia a los efectos de la competición de otra especie. Si existen dos especies (o cepas) en el mismo recipiente, el nuevo término a añadir en la ecuación de la especie 1 ha de tener como características: 1) ser un término negativo; 2) incluir la capacidad de almacenamiento de la especie 1; 3) incluir el coeficiente de competición $\alpha_{1,2}$.

La ecuación se convierte en:

$$\frac{dN_1}{dt} = r_1 \bullet N_1 \bullet \left(1 - \frac{N_1}{k_1}\right) - r_1 \bullet N_1 \bullet \left(\alpha_{1,2} \bullet \frac{N_2}{k_1}\right) \quad (11.11)$$

donde el binomio $r_1 \bullet N_1$ representa el crecimiento geométrico, el primer paréntesis los efectos de la especie 1 sobre sí misma y el segundo paréntesis los efectos de la especie 2 sobre la especie 1.

Si ampliamos las cepas en convivencia a tres, la ecuación resulta:

$$\frac{dN_2}{dt} = r_2 \bullet N_2 \bullet \left(1 - \frac{N_2}{k_2} - \alpha_{2,1} \bullet \frac{N_1}{k_2} - \alpha_{2,3} \bullet \frac{N_3}{k_2}\right) \quad (11.12)$$

Los signos negativos de la ecuación 11.12 indican la situación de competición. No obstante, se puede alterar la relación entre las cepas y modificar la naturaleza de esta interacción cambiando los signos de la siguiente manera: competición $(-, -)$; parasitismo $(-, +)$; o simbiosis $(+, +)$. En la ejecución de la última simulación se mantienen los signos negativos de la ecuación 11.12 cuyo modelado en Simulink se ilustra en la figura 11.24.

La figura 11.24b representa la ecuación en la cepa de fluorescencia roja, la cual se diferencia de las demás únicamente en el valor de las constantes. Para la simulación, estos valores han sido definidos de manera que se consiga una competición

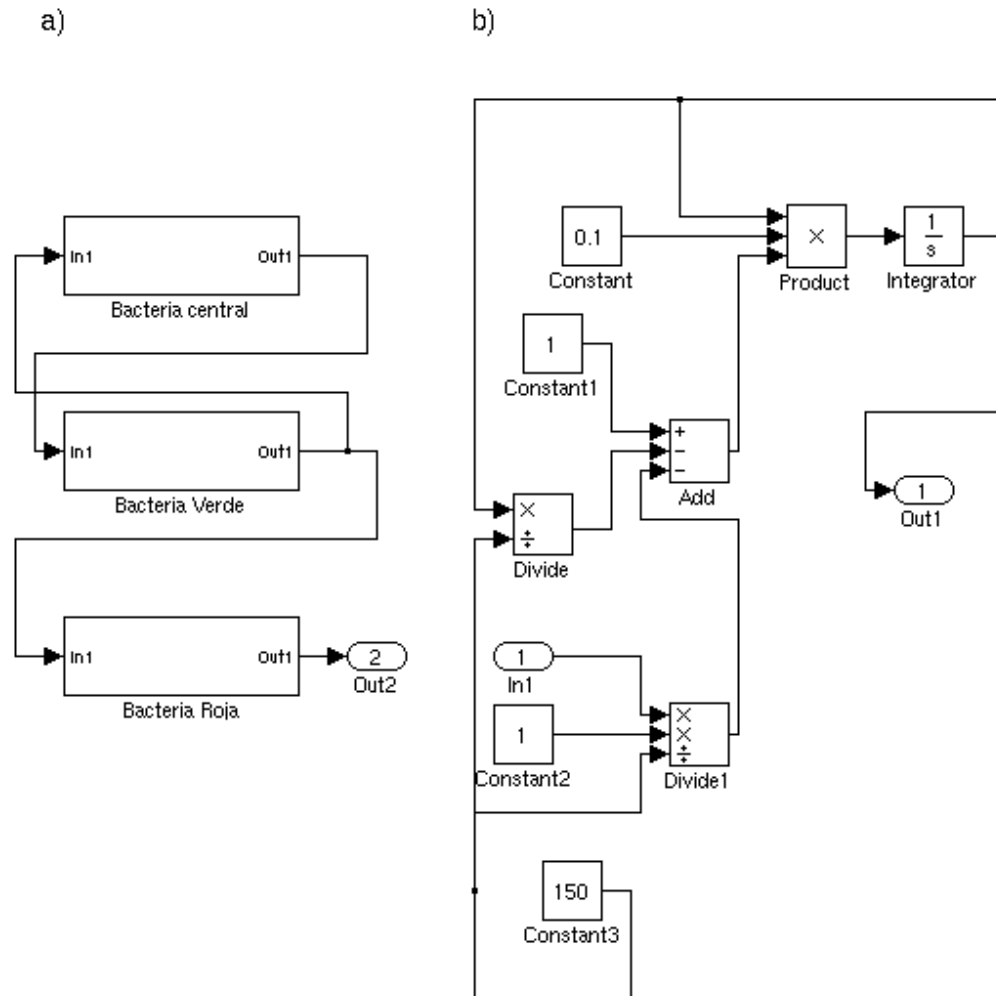


Figura 11.24: Función de crecimiento condicionada

excesiva entre las cepas con el fin de exagerar el escenario y probar el oscilador en condiciones extremas. En la figura 11.24a se observan las relaciones de competición entre el servidor y los clientes del biodispositivo que dan lugar a la gráfica 11.25a en la que dos cepas acaban por desaparecer en espacio dedicado al cultivo de las cepas.

Contemplando el gráfico 11.25 correspondiente al comportamiento oscilatorio en esta última simulación, se observa cómo el mecanismo acaba por apagarse y dejar de funcionar. Esto es debido a que una o más cepas decrecen excesivamente en número de células y, como resultado, no se producen las moléculas AHL necesarias para el correcto funcionamiento de la oscilación.

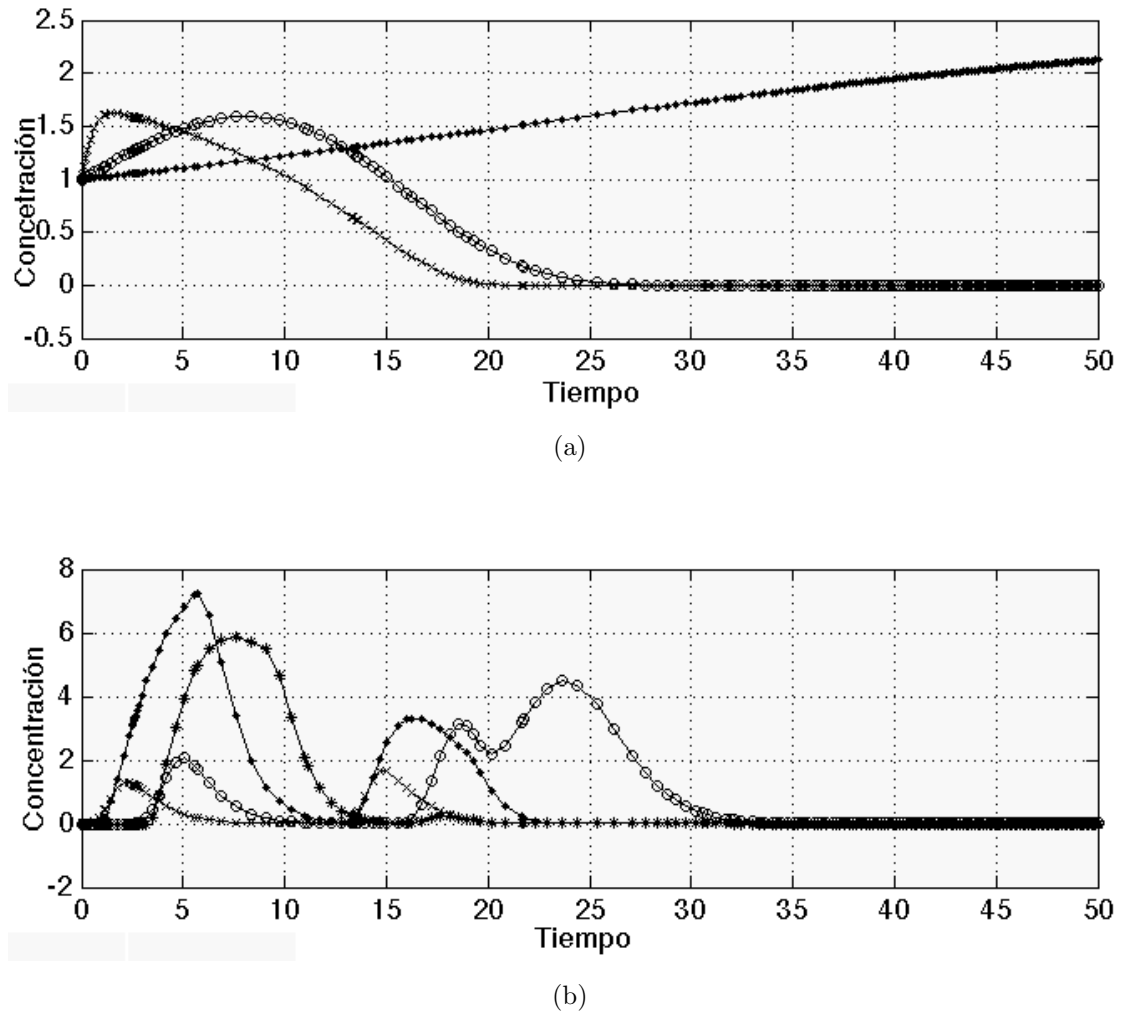


Figura 11.25: Simulación dinámica en situación de competición. a) Competición de las tres cepas. b) Oscilación resultante. La oscilación mide la concentración de las distintas moléculas AHL en solución: AHLr(\times), AHLg(\circ), AHLs'(\bullet), AHLs($*$). La fluorescencia verde coincide con la expresión de AHLs y la roja con AHLs'.

11.4.3. Discusion

Antes de mencionar las principales líneas de investigación futuras derivadas de este diseño, cabe destacar un factor relevante a la viabilidad de, no solo este último, sino los dos últimos desarrollos. El hecho de diseñar cada bacteria de tal forma que la totalidad de la información genética que guía el comportamiento de éstas se pueda codificar en el cromosoma bacteriano en forma de instrucciones computacionales,

hace innecesario el uso de vectores o plásmidos en la comunidad heterogénea.

Al no usar los procesos característicos de la transferencia horizontal de información genética, se reduce considerablemente la carga de proceso de la bacteria y su presión metabólica. Esto hace que se pueda apartar ese mecanismo del diseño de las células mínimas que lleven a cabo mecanismos de este estilo, con la consiguiente reducción de complejidad.

Como líneas futuras de investigación que se concluyen de este capítulo, destacan:

- Afrontar el crecimiento bacteriano y las relaciones entre cepas como comportamientos a controlar, siendo estos la barrera a superar con el fin de diseñar dispositivos multicelulares robustos basados en la cooperación multi-cepa.

El crecimiento representa a los dispositivos celulares lo que el ruido genético intrínseco representa a los dispositivos unicelulares y las relaciones de competición, parasitismo o simbiosis pueden desembocar en situaciones impredecibles en el biodispositivo sintético diseñado. Es importante notar que no sólo es conveniente controlar las situaciones anteriores para que no se produzcan, también puede ser ventajoso sacar partido de ellas provocando su aparición en momentos determinados de la computación.

El desarrollo de células sintéticas, capaces de aprovechar todo el rango de relaciones entre cepas aplicadas a alcanzar objetivos concretos, abre las puertas a un campo de investigación con un potencial inmenso.

- La utilización de promotores híbridos sintéticos es una herramienta muy potente para el modelado de estos computadores bacterianos. Hasta no hace mucho tiempo y salvo alguna excepción aislada, los diseños de circuitos sintéticos han utilizado únicamente para su desarrollo promotores constitutivos, inducibles o represibles.

Sin embargo, las modificaciones de las regiones del ADN que codifican los promotores en búsqueda de distintas funcionalidades en la eficiencia de la

transcripción, añaden características específicas muy útiles a estos elementos clave de la ingeniería molecular. En el diseño del oscilador aquí propuesto se observa cómo los promotores híbridos pueden desempeñar tareas que deriven en ventajas a priori desconocidas como la flexibilidad en el diseño de los umbrales de señalización.

- La búsqueda de diseños que apliquen la funcionalidad de un oscilador genético puede ser la conclusión o línea de investigación futura más relevante no sólo de este trabajo sino de todos los desarrollos de osciladores que en los últimos años se han publicado.

Al emular las características de la computación tradicional, la señal de reloj oscilatoria que se consigue en la comunidad heterogénea de bacterias sirve como señal de sincronización y coordinación de los procesos que en ella ocurren. Debido al carácter altamente difuso de los procesos moleculares no es posible asignar períodos de tiempo constantes entre procesos. Es decir, no se puede afirmar que entre dos respuestas celulares específicas transcurre siempre x tiempo ya que las bacterias son seres vivos que, salvo coincidencia, no realizan una función dos veces a la misma velocidad.

Es por esto que el oscilador genético se basa en otra escala de medida síncrona y atemporal de manera que se pueda medir el intervalo entre las dos respuestas celulares en ciclos de oscilación.

Parte IV

Conclusiones y líneas de investigación futuras

Conclusiones

La computación con bacterias está aún en su infancia y los desarrollos especificados en esta Tesis Doctoral ofrecen, además de importantes propuestas y algoritmos, varias líneas de investigación a explorar. La más importante de ellas es, de manera obvia, la sincronización de cepas bacterianas para la cooperación de funcionalidades diversas en un único comportamiento emergente.

En la Naturaleza se pueden encontrar *biofilms* o comunidades de bacterias en las que cientos de especies distintas consiguen colaborar para lograr la estabilidad y robustez de su población. Sin embargo, aún resulta muy complicado cultivar en laboratorio cepas diferentes de la misma especie en una misma comunidad sin que la convivencia resulte inestable y peligrosa para la función de alguna de ellas. Es para dar respuesta a este problema que la presente Tesis centra sus desarrollos en el diseño de arquitecturas de comunicaciones para comunidades de bacterias multi-cepa.

La comunidad heterogénea diseñada en el capítulo 9 resuelve un problema complejo utilizando la conjugación bacteriana como protocolo de comunicaciones fundamental. La idea de diferenciar las instrucciones computacionales almacenadas en el cromosoma bacteriano -en analogía a las instrucciones ensamblador almacenadas en memoria- y los conjuntos de datos almacenados en vectores plasmídicos representa un interesante enfoque del empleo de la información codificada genéticamente.

Los desarrollos de los capítulos 10 y 11 parten también del objetivo de conseguir un comportamiento emergente de la mezcla de varias cepas funcionalmente diferentes en una misma comunidad, diferenciándose del anterior en el protocolo de comunicaciones empleado. El *quorum sensing* sirve aquí como mecanismo fundamental para alcanzar el comportamiento esperado: un circuito lógico XOR y un oscilador poblacional respectivamente. De este último cabe destacar el empleo de promotores híbridos como fuente de robustez y medio para la reducción del ruido genético generado en el oscilador.

De los tres ejemplos anteriores se enfatiza, aparte del carácter autónomo de

la computación, la metodología seguida en el diseño de cada bacteria o componente de los desarrollos: la independencia total en el funcionamiento del componente. Es decir, una bacteria es diseñada para llevar a cabo una función específica por si sola sin la necesidad del sistema completo, potenciándose así el carácter de reutilización de los componentes. De esta manera la cepa AND perteneciente al biocircuito XOR del capítulo 10 puede utilizarse en la construcción de otro dispositivo, así como la bacteria servidora del oscilador del capítulo 11 puede ser de útil empleo en otro diseño que nada tenga que ver con el oscilador.

Esta Tesis, lejos de representar el fin de una linea de investigación específica, significa un paso más en el comienzo de una ciencia interdisciplinar que sin duda emergerá con fuerza durante la primera mitad del siglo XXI. Como tal, las conclusiones de este trabajo apuntan direcciones concretas como la creación de dispositivos multicelulares con rudimentarios mecanismos de toma de decisiones. La bacteria servidora del oscilador poblacional ya cuenta con la capacidad básica de diferenciar varios estados en la entrada y actuar en consecuencia pero no supedita su actividad a las necesidades del biodispositivo global, lo cual supondría una toma importante de decisiones y probaría la posibilidad de *programar* en bacterias comportamientos hasta ahora restringidos a formas de vida superior.

Las propuestas de codificación de la información presentadas en los capítulos 3 y 5 ofrecen también lineas futuras de investigación en lo referente a lo que podríamos llamar programación a bajo nivel e importantes conclusiones en cuanto a la potencia del uso de la información. Destaca en este apartado la propuesta de lectura analógica en la salida de los algoritmos *in-vivo* la cual presenta un importante campo de aplicación en la construcción de biosensores.

Conclusions

Bacterial computing is an emerging research area which is still in its infancy. The developments specified in the present Thesis suggest, apart from original algorithms, some open questions or future research lines. The most important of them, as introduced by the models proposed, is the synchronisation of bacterial strains with different functionality in order to achieve populations with a unique behaviour.

In nature there are *biofilms* or bacterial populations in which hundreds of bacterial species constitute a robust and stable community by improving their cooperative skills. However, it is still difficult to cultivate engineered strains in artificial mediums without making the coexistence being dangerous or unstable for the different cells. In order to come up with solutions to tackle that problem, the present Thesis focuses its developments in the design of communication architectures for multi-strain bacterial communities based on an analogy with computational objectives and methods.

The heterogeneous community designed in chapter 9 can solve a typical combinatorial problem (SAT) by using bacterial conjugation as the basic communication protocol. The idea of drawing a distinction between computational instructions which are stored in the chromosome of the cells -in a clear reference to the memory of a processor- and the data sets stored in plasmid vectors so that they can *travel* from one processor to another, represents an interesting approach to the use of genetically encoded information.

The developments shown in chapters 10 and 11 also aim at building a biosystem formed from different functional bacterial strains where a unique emerging behaviour is achieved. The main difference between these ones and the SAT-solver community is that the communication protocol used is the *quorum sensing* mechanism which is altered here to reach the final expected functionality: a XOR logic circuit and a population-based oscillator respectively. From the latest it is significant to highlight the utilisation of specific building blocks called hybrid promoters in order to reduce the genetic noise in oscillation cycles and help the robustness of the model.

From the three previous models it is emphasised, apart from the autonomous behaviour of the computations (no human assistance), the methodology used to design each bacterium or functional component: the self-sufficient function of every part. That is to say, bacteria are designed to perform a specific role by itself without the need of being part of a machine so that the reusability feature of the parts is accentuated. For instance, the AND strain belonging to the XOR biocircuit of chapter 10 can be used in the construction of another logic circuit and the server bacteria integrated in the oscillator of chapter 11 can be useful in other biosystems whose objective is completely different.

This Thesis, far from representing the end of a specific research line, represents a significant contribution to the beginnings of an interdisciplinary field whose importance increases exponentially every year. The conclusions of this work point at specific directions for future research like the model of biosystems with rudimentary decision making processes. The server bacteria of the population-based oscillator already owns a basic capacity to distinguish several states in its input and decide which action to perform but it doesn't subordinate that action to the necessities of the population which would indicate an important decision making feature and could corroborate the possibility to *program* bacteria with behaviours restricted to more complex living beings.

The proposals referred to information encoding presented in chapters 3 y 5 also offer future research lines in what could be denoted as *low level programming* and important conclusions alluding the use of information. About this part of the Thesis is highlighted the proposition of analogue reading in bacterial output responses because it represents a relevant paradigm whose methodology can be applicable in building biosensors with a wider range of expression.

Publicaciones del Autor

contacto: agmoreno@gcn.upm.es

- [GoM10]* **Goñi-Moreno A**, Amos M. *A population-based microbial oscillator*. (in submission) (2010)
- [GoM10]* **Goñi-Moreno A**, Cordero P, Arteta A, Castellanos J. *Analog output for bacterial computing*. Theoretical Computer Science (submitted). (2010)
- [GoM10]* **Goñi-Moreno A**, Amos M. *Microbial computation of solutions to combinatorial problems*. (in submission) (2010)
- [GoM10]* **Goñi-Moreno A**, Amos M. *Engineered microbial communication for population-level behaviour*. 12th International Conference on the Synthesis and Simulation of Living Systems, Denmark. (accepted). (2010)
- [GoM10]* **Goñi-Moreno A**, Redondo-Nieto M, Arroyo F, Castellanos J. *Biocircuit design through engineering bacterial logic gates*. Natural Computing, DOI: 10.1007/s11047-010-9184-2. (2010)
- [Art10] Arteta A, Arroyo A, **Goñi-Moreno A**. *Calculating maximal multisets of objects by using RAM as support*. 15th International Symposium on Artificial Life and Robotics AROB'10. Japan. (OS):28-31. (2010)
- [GoM09]* **Goñi-Moreno A**, Cordero P. *Conditional development of bacterial based swarm intelligent systems and real biological manipulation*. Proc of Computer Science and Technologies conference CSIT'O9. 222-225. (2009)
- [Cor09]* Cordero P, **Goñi-Moreno A**, Castellanos J. *Using directed evolution techniques to solve hard combinatorial problems*. Proc of Computer Science and Technologies conference CSIT'O9. 225-229.(2009)
- [Rio09] Rio T, Cisneros FJ, **Goñi-Moreno A**, Castellanos J. *Simulation of a DNA denaturation process*. Proc. of the 7th ACS/IEEE International Conference on Computer Systems and Applications. 737-742. (2009)

-
- [DeL09] De la Peña A, Cisneros FJ, **Goñi-Moreno A**, Castellanos J. *DNA electrophoresis simulation*. Lecture Notes in Computer Science. 5518: 585-588. (2009)
- [GoM09] **Goñi-Moreno A**, Cordero P. *Global memory structure for ant colony optimization algorithms*. Intelligent Engineering. 3:116-122. (2009).
- [Art09] Arteta A, **Goñi-Moreno A**, Castellanos J. *Analysis of P-Systems under a multiagent system perspective*. Intelligent Processing. 3: 117-128.(2009)
- [GoM08]* **Goñi-Moreno A**, Cisneros FJ, Cordero P, Castellanos J. *A DNA codification for genetic algorithms simulation*. Advance Research in Artificial Intelligence. 2:69-74. (2008)
- [GoM08]* **Goñi-Moreno A**. *Solving Travelling Salesman Problem in a simulation of genetic algorithms with DNA*. Information Theories and Applications. 15(4): 357-363. (2008).
- [BaR08] Baños R, Cordero P, **Goñi-Moreno A**, Castellanos J. *Simulation of a DNA renaturalization process*. International Conference on Metaheuristics and Nature Inspired Computing META'08. (2008)
- [GoM07]* **Goñi-Moreno A**. *Crossover operator in DNA simulation of genetic algorithms*. Proc. of the 5th International Conference Information Research and Applications I-TECH'07. 1: 74-79. (2007)

* Publicación derivada de esta Tesis Doctoral

Bibliografía

- [Adl94] Adleman L. *Molecular computation of solutions to combinatorial problems*. Science. 266:1021-1024. (1994)
- [Alb09] Alberghini S, Polone E, Corich V, Carlot M, Seno F, Trovato A, Squartini A. *Consequences of relative cellular positioning on quorum sensing and bacterial cell-to-cell communication*. FEMS Microbiol. Lett. 292: 149-161. (2009)
- [Amo09] Amos M. *Bacterial Computing*. Encyclopedia of Complexity and Systems Science. 417-426. (2009)
- [And07] Anderson JC, Voigt CA, Arkin AP. *Environmental signal integration by a modular AND gate*. Molecular Systems Biology. 3:133. (2007).
- [Bac99] Back T, Kok JN, Rozemberg G. *Cross-fertilization between evolutionary computation and DNA-based computing*. Proc. 1999 Congress on Evolutionary Computation CEC. 2: 980-987. (1999)
- [Bae01] Baer A, Bode J. *Coping with kinetic and thermodynamic barriers: RMCE, an efficient strategy for the targeted integration of transgenes*. Current Opinion in Biotechnology. 12(5): 473-480. (2001)
- [Bal10] Ball DA, Lux MW, Graef RR, Peterson MW, Valenti JD, Dileo J, Peccoud J. *Co-design in synthetic biology: a system-level analysis of the development of an environmental sensing device*. Pacific Symposium on Biocomputing. 15: 388-396. (2010)

- [Bar04] Barron JA, Rosen R, Jones-Meehan J, Spargo BJ, Belkin S, and Ringeis BR. *Biological laser printing of genetically modified Escherichia coli for biosensor applications*. Biosens. Bioelectron. 20(2):246-252, (2004)
- [Bas05] Basu S, Gerchman Y, Collins C, Arnold FH, Weiss R. *A synthetic multicellular system for programmed pattern formation*. Nature 434: 1130-1134. (2005)
- [Bau09] Baumgardner J, Acker K, Adefuye O, Crowley ST, DeLoache W, Dickson JO, Heard L, Martens AT, Morton N, Ritter M, Shoecraft A, Treece J, Unzicker M, Valencia A, Waters M, Campbell AM, Heyer LJ, Poet JL, Eckdahl TT. *Solving a Hamiltonian Path Problem with a bacterial computer*. Journal of Biological Engineering. 3:11. (2009)
- [Ben01] Beneson Y, Paz-Elizur T, Adar R, Keinan E, Livneh Z, Shapiro E. *Programmable and autonomous computing machine made of biomolecules*. Nature, 414:430-434. (2001)
- [Bje06] Bjerketorp J, Hekansson S, Belkin S, and Jansson JK. *Advances in preservation methods: keeping biosensor microorganisms alive and active*. Curr. Opin. Biotechnol. 17(1):43-49, (2006)
- [Boe83] Boer HA, Comstock LJ, Vasser M. *The tac promoter: a functional hybrid derived from the trp and lac promoters*. Proc Natl Acad Sci U.S.A. 80(1):21-25, (1983)
- [Bre07] Brenner K, Karig DK, Weiss R, Arnold FH. *Engineered bidirectional communication mediates a consensus in a microbial biofilm consortium*. PNAS 104(44): 17300-17304. (2007)
- [Bre08] Brenner K, You L, Arnold FH. *Engineering microbial consortia: a new frontier in synthetic biology*. Trends Biotechnol. 26(9):483-489. (2008)
- [Bur06] Burmolle M, Webb JS, Rao D, Hansen LH, Sorensen SJ, Kjelleberg S. *Enhanced biofilm formation and increased resistance to antimicrobial agents and bacterial invasion are caused by synergistic interactions in multispecies biofilms*. Applied and Environmental Microbiology, 72(6): 3916-3923. (2006)

- [Cha09] Chandran D, Bergmann FT, Sauro HM. *TinkerCell: modular CAD tool for synthetic biology*. Journal of Biological Engineering 3:19doi: 10.1186/1754-1611-3-19. (2009)
- [Cla98] Clark A. *Twisted Tales: Casual Complexity and Cognitive Scientific Explanation*. Minds and Machines, 8:79-99. (1998)
- [Col09] Colmenero C, Colomo B, Conejo A, Cutuli MT. *Vida social de las bacterias: aquí no hay quien viva; Bacterial social life: anyone can't be lives here*. RCCCCV 3(2): 235-243. (2009)
- [Cos95] Costerton J.W., Lewandowski Z., Caldwell D.E., Korber D.R., and Lappin-Scott H.M. *Microbial Biofilms*. Annu. Rev. Microbiol. 49:711-745. (1995)
- [Cos99] Costerton J.W., Stewart P.S., and Greenberg E.P. *Bacterial biofilms: a common cause of persistent infections*. Science. 284:1318-1322. (1999)
- [Cui07] Cui B, Konstantinidis S. *DNA coding using the subword closure operation*. Proc. of the 13th Meeting on DNA Computing DNA13. 65-74. (2007)
- [Dan09] Danchin A. *Bacteria as computers making computers*. FEMS Microbiol. Rev. 33: 3-26. (2009)
- [Dan10] Danino T, Mondragón-Palomino O, Tsimring L, Hasty J. *A synchronized quorum of genetic clocks*. Nature 463: 326-330. (2010)
- [Dor99] Dorigo M, Di Caro G, Gambardella LM. *Ant Algorithms for Discrete Optimization*. Artificial Life, 5(2):137-172, (1999)
- [Dru07] Drubin DA, Way JC, Silver PA. *Designing biological systems*. Genes Dev. 21: 242-254. (2007)
- [Eit08] Eiteman MA, Lee SA, Altman E. *A co-fermentation strategy to consume sugar mixture effectively*. Journal of Biol. Eng. 2:3. (2008)
- [Elo00] Elowitz MB, Leibler S. *A synthetic oscillatory network of transcriptional regulators*. Nature, 403: 335-338. (2000)
- [Erl89] Erlich HA. *PCR technology. Principles and applications for DNA amplification*. Stockton Press. New York USA. 246p. (1989)

- [Fey61] Feynman RP. *There is plenty of room at the bottom*. Miniaturization 282-296. (1961)
- [Fra07] Fraser C, Hanage WP, Spratt BG. *Recombination and the Nature of Bacterial Speciation*. Science, 315(5811):476-480. (2007)
- [Gar00] Gardner TS, Cantor CR, Collins JJ. *Construction of a genetic toggle switch in Escherichia coli*. Nature, 403: 339-342. (2000)
- [Gog89] Gogarten JP, Kibak H, Dittrich P, Taiz L, Bowman EJ, Bowman BJ, Manolson MF, Poole RJ, Date T, Oshima T. *Evolution of the vacuolar H⁺-ATPase: implications for the origin of eukaryotes*. PNAS, 86(17): 6661-6665, (1989)
- [Gog02] Gogarten JP, Doolittle WF, Lawrence JG. *Prokaryotic Evolution in Light of Gene Transfer*. Molecular Biology and Evolution 19:2226-2238, (2002)
- [Gua96] Guarnieri F, Fliss M, Bancroft C. *Making DNA add*. Science, 273:220-223. (1996).
- [Han97] Hannon BM, Ruth M. *Modeling dynamic biological systems*. New York: Springer, (1997)
- [Hay08] Haynes KA, Broderick ML, Brown AD, Butner TL, Dickson JO, Harden WL, Heard LH, Jessen EL, Malloy KJ, Ogden BJ, Rosemond S, Simpson S, Zwack E, Campbell AM, Eckdahl TT, Heyer LJ, Poet JL. *Engineering bacteria to solve the Burnt Pancake Problem*. Journal of Biological Engineering. 2:8. (2008)
- [Hea87] Head T. *Formal language theory and DNA: An analysis of the generative capacity of specific recombinant behaviors*. Bulletin of Mathematical Biology 49(6): 737-759. (1987)
- [Hea00] Head T, Rozemberg G, Bladergroen RS, Breek CKD, Lommerse PHM, Spaink HP. *Computing with DNA by operating on plasmids*. Biosystems. 57(2): 87-93. (2000)

- [Hei95] Heim R, Cubitt A, Tsien R. *Improved green fluorescence*. Nature, 375(6516): 663-664. (1995)
- [Hen05] Henkel CV, Bladergroen RS, Balog CI, Deelder AM, Head T, Rozemberg G, Spalink HP. *Protein output for DNA computing*. Natural Computing. 4(1): 1-10. (2005)
- [Hoo06] Hooshangi S, and Weiss R. *The effect of negative feedback on noise propagation in transcriptional gene networks*. Chaos. 16(2):26108, (2006)
- [Hue98] Huerta AM, Salgado H, Thieffry D, Collado-Vides J. *RegulonDB: a database on transcriptional regulation in Escherichia coli*. Nucleic Acids Res. 26(1):55-59. (1998)
- [Hug88] Hughes KT, Youderian P, Simon MI. *Phase variation in Salmonella: analysis of Hin recombinase and hix recombination site interaction in vivo*. Genes Development. 2: 937-948. (1988)
- [Jay08] Jayaraman A, Wood TK. *Bacterial quorum sensing: signals, circuits, and implications for biofilms and disease*. Annu. Rev. Biomed. Eng.10:145-167. (2008)
- [Jun96] Gu J, Purdom PW, Franco J, Wah BW. *Algorithms for the Satisfiability (SAT) Problem: A Survey*. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 19-152. (1996)
- [Kar05] Kari L, Konstantinidis S, Sosik P. *Preventing undesirable bonds between DNA codewords*. LNCS 3384: 182-191. (2005)
- [Kel06] Keller L, Surette MG. *Communication in bacteria: an ecological and evolutionary perspective*. Nat. Rev. Microbiol. 4: 249-258. (2006)
- [Kid07] Kidykas K. *Biobrick type modeling*. iGEM Glasgow (2007)
- [Kur07] Kuramitsu HK, He X, Lux R, Anderson MH, Shi W. *Interspecies Interactions within Oral Microbial Communities*. Microbiol. and Molecular Biology Reviews. 71(4):653-670. (2007)

- [Lam08] Lam FH, Steger DJ, O'Shea EK. *Chromatin decouples promoter threshold from dynamic range*. Nature, 453: 246-250, (2008)
- [Lan09] Land M, Lapidus A, Mayilraj S, Chen F, Copeland A, Glavina del Rio T, Nolan M, Lucas S, Tice H, Cheng J-F, Chertkov O, Bruce D, Goodwin L, Pitluck S, Rohde M, Goeker M, Pati A, Ivanova N, Mavromatis K, Chen A, Palaniappan K, Hauser L, Chang Y-J, Jeffries CC, Brettin T., Detter JC, Han C, Chain P, Tindall BJ, Bristow J, Eisen JA, Markowitz V, Hugenholtz P, Kyrpides NC, Klenk H-P. *Complete genome sequence of Actinosynnema mirum type strain (101T)*. Stand. Genomic Sci. 1:46-53. (2009)
- [Lev09] Levin BR, Cornejo OE. *The population and evolutionary dynamics of homologous gene recombination in bacteria*. PLoS Genet. 5(8):e1000601. (2009)
- [Loe09] Loeckx J, Gielen GGE. *Generic and accurate whitebox behavioral model for fast simulation of analog effects in nanometer CMOS digital logic circuits*. IEEE Transactions. 51(2):351-357. (2009)
- [Mat97] Mathee K, McPherson CJ, and Ohman DE. *Posttranslational control of the algT (algU)-encoded sigma22 for expression of the alginate regulon in Pseudomonas aeruginosa and localization of its antagonist proteins MucA and MucB (AlgN)*. J. Bacteriol. 179(11):3711-3720, (1997)
- [Mie75] Miescher F. *Ueber die chemische Zusammensetzung der Eiterzellen*. Med. Chem. Unders. 4: 441-460. (1875)
- [Mon09] Monteiro SMD, Goldbarg EFG, Goldbarg MC. *A plasmid based transgenic algorithm for the biobjective minimum spanning tree problem*. Lecture Notes in Computer Science 5482: 49-60. (2009)
- [Mor92] Morowitz HJ. *Beginnings of Cellular Life: Metabolism*. Recapitulates Biogenesis. Yale University Press, Binghamton, NY. (1992)
- [Moy09] Moya A, Gil R, Latorre A, Juli Peret J, Garcillán-Barcia MP, de la Cruz F. *Toward minimal bacterial cells: evolution vs design*. FEMS Microbiol. Rev. 33:225-235. (2009)

- [Mun03] Münch R, Hiller K, Barg H, Heldt D, Linz S, Wingender E, Jahn D. *PRO-DORIC: prokaryotic database of gene regulation*. Nucleic Acids Res. 31(1):266-269. (2003)
- [Ncl77] Ncalson KH. *Autoinduction of bacterial luciferase. Occurrence, mechanism and significance*. Arch. Microbiol. 112:73-79. (1977)
- [Neu66] Neumann JV. *Theory of Self-Reproducing Automata*. University of Illinois Press. Edited and completed by Arthur W. Burks. (1966)
- [Nev09] Nevin KP, Kim BC, Glaven RH, Johnson JP, Woodard TL, Methé BA, DiDonato RJ, Covalla SF, Franks AE, Liu A, Lovley DR. *Anode biofilm transcriptomics reveals outer surface components essential for high density current production in Geobacter sulfurreducens fuel cells*. PLoS ONE 4(5):e5628. (2009)
- [Nor08] Norris V, Zemirline A, Amar P, Ballet P, Ben Jacob E, Bernot G, Beslon G, Fanchon E, Giavitto JL, Glade N, Greussay P, Grondin Y, Foster JA, Hutzler G, Kepes F, Michel O, Misevic G, Molina F, Signorini J, Stano P, and Thierry A. *From bioputing to bactoputing: computing with bacteria*. In: EDP Science (ed) Modelling Complex Biological Systems in the Context of Genomics. pp 123-150, (2008)
- [Per09] Pérez-Mendoza D, de la Cruz F. *Escherichia coli genes affecting recipient ability in plasmid conjugation: Are there any?*. BMC Genomics. 10:71. (2009)
- [Put84] Putman RJ, Wratten SD. *Principles of Ecology*. Berkeley and Los Angeles: University of California Press, (1984)
- [Saka05] Sakamoto K, Yamamura M, Someya H. *Toward wet implementation of genetic algorithm for protein engineering*. LNCS 3384 DNA Computing DNA10. 308-318. (2005)
- [San77] Sanger F, Nicklen S, Coulson AR. *DNA sequencing with chain-terminating inhibitors*. Proc. Natl. Acad. Sci. USA 74(12):5463-5467. (1977)
- [Sch03] Scholz F, Schroumler U. *Bacterial batteries*. Nature Biotechnology 21:1151-1152. (2003)

- [Sha05] Shaner N, Steinbach P, Tsien R. *A guide to choosing fluorescent proteins*. Nat. Methods, 2(12): 905-909. (2005)
- [Sim05] Simpson ML, Saylor GS, Fleming JT, Sanseverino J, Cox CD. *The device science of whole cells as components in microscale and nanoscale systems*. Book chapter (5). Cellular Computing. Ed. Martyn Amos. Series in Systems Biology, Oxford University Press. (2005)
- [Soh09] Sohka T, Heins RA, Phelan RM, Greisler JM, Townsend CA, Ostermeier M. *An externally tunable bacterial band-pass filter*. PNAS, 106(25): 10135-10140. (2009)
- [Vig03] Vigoda B. *Analog logic: Continuous-Time analog circuits for statistical signal processing*. Doctoral Thesis. Massachusetts Institute of Technology (2003)
- [Wak05] Wakabayashi K, Yamamura M. *The Enterococcus faecalis information gate*. Book chapter (6). Cellular Computing. Ed. Martyn Amos. Series in Systems Biology, Oxford University Press. 109-118 (2005)
- [Waka05] Wakabayashi K, Yamamura M. *A design for cellular evolutionary computation by using bacteria*. Natural Computing. 4(3): 275-292. (2005)
- [Wan01] Wang L, Hall JG, Lu M, Liu Q, Smith LLM. *A DNA computing readout operation based on structure-specific cleavage*. Nature Biotechnol. 19:1053-1059. (2001).
- [Wat53] Watson J, Crick F. *A structure for Desoxyribose Nucleic Acid*. Nature 171(1): 737. (1953)
- [Wei99] Weiss R, Homsy G, Knight TF.Jr. *Toward in-vivo digital circuits*. DIMACS Workshop on Evolution as Computation. Springer-Verlag. (1999)
- [Wes06] West SA, Griffin AS, Gardner A. *Social semantics: altruism, cooperation, mutualism, strong reciprocity and group selection*. Europ. Soc. Evol. Biol. 20:415-432. (2006)
- [Wes07] West SA, Diggle SP, Buckling A, Gardner A, Griffin AS. *The social lives of microbes*. Annu. Rev. Ecol. Evol. Syst. 38:53-77. (2007)

- [Wil08] Wilson D, Charoensawan V, Kummerfeld SK, Teichmann SA. *DBD-taxonically broad transcription factor predictions: new content and functionality*. Nucleic Acids Res. 36(Database issue):D88-92. (2008)
- [Woo01] Wood D, Chen J, Antipov E, Lemieux B, De N. *A design for DNA computation of the OneMax problem*. Soft Comput. 5(1): 19-24. (2001)
- [YiH09] Yi H, Nevin KP, Kim BC, Franks AE, Klimes A, Tender LM, Lovley DR. *Selection of a variant of Geobacter sulfurreducens with enhanced capacity for current production in microbial fuel cells*. Biosens Bioelectron 24(12):3498-3503. (2009)